



Programma  
**Aan de slag met de  
Omgevingswet**

**Functionele documentatie**

**Ozon API**

**Omgevingsdocumenten Verbeelden v3**

*Versie 21 november 2022*

## Inhoud

1	Inleiding.....	3
1.1	Ozon.....	4
1.2	DSO-viewers.....	4
2	Standaarden en informatiemodellen.....	5
2.1	Informatiemodel Omgevingswet (IMOW).....	5
2.2	Informatiemodel Officiële Publicatie (IMOP).....	5
2.3	Presentatiemodel.....	5
2.4	REST, HTTP, JSON, HAL.....	5
3	Toegankelijkheid.....	6
3.1	Omgevingsdocumenten Verbeelden API.....	6
4	Globale procesbeschrijving.....	7
4.1	Verbeelden API en PDOK Tile API in samenhang.....	7
4.2	Genereren vectortiles.....	8
4.3	Aandachtspunten bij genereren vectortiles.....	9
4.4	Omgevingsdocumenten Tiles API (WMTS).....	11
5	Endpoints en schema's.....	12
5.1	/symbolen/mapbox/_zoek.....	12
5.2	/symbolen.....	12
5.3	/spritesheet.....	13
5.4	Tijdreizen.....	14
5.5	/health en /app-info.....	14
6	Zoekobjecten.....	15

# 1 Inleiding

Omgevingsdocumenten van bevoegd gezagen (Rijk, Provincies, Gemeenten en Waterschappen) worden door de Landelijke Voorziening Bekendmaken en Beschikbaar stellen (LVBB) bekendgemaakt en beschikbaar gesteld.

De Omgevingsdocumenten Verbeelden API (hierna: Verbeelden API), gebaseerd op CIM-OW<sup>1</sup> en toegespitst op het verbeelden van gegevens uit Omgevingsdocumenten in een kaartbeeld. In het DSO heeft de Locatie een geometrie, en verwijzen objecttypen zoals een Omgevingsnorm, een Gebiedsaanwijzing, een Regel enz. naar deze Locatie als zijnde de locatieaanduiding van de objecten. De objecten hebben zelf geen geometrie. Voor de verbeelding op de kaart worden de Locaties en de andere objecttypen bij elkaar gebracht en getoond.

- De Locaties met hun geometrie worden ontsloten door de Omgevingsdocumenten Tiles API (WMTS), via Publieke Dienstverlening op de Kaart (PDOK)<sup>2</sup>. Deze API zorgt voor de geometrie ('waar' moet de visualisatie op de kaart staan).
- De te verbeelden informatie van de OW-objecten, en de bijbehorende symbolisatiecodes en styles, worden via de Omgevingsdocumenten Verbeelden API ontsloten. De Verbeelden API verzorgt de objectverbeelding van de overige CIM-OW-objecten.

De Verbeelden en Tile API's worden bevraagd met behulp van OW-gegevens (identificaties) van objecten die verbeeld moeten worden op de kaart. Aan de hand van de identificaties van bijvoorbeeld een omgevingsnorm, gebiedsaanwijzing of kaartobject/kaartlaag, wordt teruggegeven hoe deze objecten verbeeld moeten worden. De Verbeelden API levert dus niet de CIM-OW objecten zelf uit, maar de verbeeldingsinformatie ten behoeve van de visualisatie op kaart van die objecten. Een uitgebreidere procesbeschrijving is te vinden in het hoofdstuk Globale Procesbeschrijving.

De Verbeelden API ondersteunt geen ruimtelijke bevragingen, in tegenstelling tot bijvoorbeeld de Omgevingsdocument Presenteren API. De Verbeelden API werkt met bevragingen met locatiedentificaties, die bijvoorbeeld via de Omgevingsdocument Presenteren API of Toepasbaar Opvragen<sup>3</sup> API opgehaald worden.

---

<sup>1</sup> [https://geonovum.github.io/TPOD/CIMOW/CIMOW\\_v2.0.0.pdf](https://geonovum.github.io/TPOD/CIMOW/CIMOW_v2.0.0.pdf)

<sup>2</sup> <https://www.pdok.nl/wmts>

<sup>3</sup> In de Toepasbaar Opvragen API krijg je middels een POST op Locatie\_zoek sinds de uitrol van PI23S3.1 alle locaties terug, niet alléén die een relatie hebben met een activiteit.

## 1.1 Ozon

Vanuit de LVBB worden geconsolideerde regelingversies van omgevingsdocumenten en gegevens daaruit (OW-objecten) geleverd aan het Digitaal Stelsel Omgevingswet Landelijke Voorziening (DSO-LV) component voor het ontsluiten van objectgeoriënteerde omgevingsdocumenten (Ozon). Ozon valideert, registreert en ontsluit deze gegevens in de vorm van (OW) informatieobjecten.

Deze informatieobjecten zijn op te vragen met de Ozon APIs. Deze APIs worden gebruikt door een aantal componenten van de DSO-LV, waaronder de DSO-viewers (Viewer Regels op Kaart en Viewer Documenten op Kaart) en de Registratie Toepasbare Regels (RTR).

## 1.2 DSO-viewers

De DSO-viewers zijn belangrijke afnemers van de Verbeelden API. Met de DSO-viewer kan geografisch (op basis van een locatie zoals een adres, perceel, een klik op de kaart of coördinaten) worden gezocht welke omgevingsdocumenten, activiteiten, normen etc. waar gelden. Bij het ontwerp van de Verbeelden API is rekening gehouden met de behoefte van de DSO-viewers om snel informatie op maat te kunnen visualiseren.

De DSO-viewers combineren de Verbeelden API met de PDOK Tiles en de Ozon Presenteren APIs.

Een voorbeeld waarbij deze combinatie duidelijk naar voren komt is bij het kaartobject. Via de Presenteren API is de identificatie van het kaartobject op te vragen, maar zonder informatie over de symboolcodes of styles. Gegevens ten behoeve van de visualisatie worden opgevraagd via de Verbeelden API en de geometrie van de te verbeelden gebieden zit in de vectortiles geleverd door de PDOK Tiles API. Wanneer een gebruiker een kaartobject op de kaart bekijkt, worden dus (minimaal) deze drie APIs bevroegd om de gevraagde kaart op in de DSO-viewer te tonen.

## 2 Standaarden en informatiemodellen

Een omgevingsdocument binnen het DSO moet voldoen aan de “standaard”. De “standaard” bestaat uit twee standaarden, namelijk het informatiemodel Omgevingswet (IMOW) en het informatiemodel officiële publicaties (IMOP).

### 2.1 Informatiemodel Omgevingswet (IMOW)

Het IMOW is gebaseerd op het Conceptueel Informatiemodel Omgevingswet (CIMOW). CIMOW is het conceptuele model waarop informatiemodellen in de ketens van het DSO gebaseerd worden.

De gegevens die deze API ontsluiten bestaan uit entiteiten zoals deze staan beschreven in het conceptueel Informatiemodel voor de Omgevingswet (CIM-OW)<sup>4,5</sup>.

### 2.2 Informatiemodel Officiële Publicatie (IMOP)

De omgevingsdocumenten die aan Ozon aangeleverd worden vanuit de LVBB zijn gemodelleerd op basis van Informatiemodel Officiële Publicaties (IMOP)<sup>6,7</sup>. IMOP is onderdeel van de Standaard Officiële Publicaties (STOP), de bovenliggende kapstok van deze standaard.

### 2.3 Presentatiemodel

De Verbeelden API levert een presentatiestijl uit volgens de Symbolenbibliotheek STOP-TPOD versie 1.3.0<sup>8</sup>.

### 2.4 REST, HTTP, JSON, HAL

De Verbeelden API is een RESTful API (Representational State Transfer) en is gebaseerd op de DSO API-strategie<sup>9</sup>. De endpoints zijn dus te bevragen aan de hand van HTTP-requests. De responses worden gegeven via HTTP-responses met JSON als inhoud. Deze antwoorden bevatten hyperlinks om verder te kunnen navigeren door de API volgens de HAL-specificatie (Hypertext Application Language). De wijze waarop dit geïmplementeerd is binnen het DSO is terug te vinden in de DSO API-strategie.

---

<sup>4</sup> <https://www.geonovum.nl/geo-standaarden/omgevingswet/STOPTPOD>

<sup>5</sup> Raadpleeg de GET/app-info van de Verbeelden API voor de actuele versie van compatibele DSO-standaarden.

<sup>6</sup> <https://koop.gitlab.io/STOP/standaard/index.html>

<sup>7</sup> Raadpleeg de GET/app-info van de Verbeelden API voor de actuele versie van compatibele DSO-standaarden.

<sup>8</sup> <https://www.geonovum.nl/documents/symbolenbibliotheekstoptpodv130zip>

<sup>9</sup> <https://iplo.nl/digitaal-stelsel/aansluiten/standaarden/api-en-uri-strategie/>

## 3 Toegankelijkheid

### 3.1 Omgevingsdocumenten Verbeelden API

Het endpoint voor de Verbeelden API via het DSO-knooppunt is vanuit verschillende omgevingen beschikbaar:

- Demo omgeving:  
<https://service.dmo.omgevingswet.overheid.nl/publiek/omgevingsdocumenten/api/verbeelden/v3>
- Productie omgeving:  
<https://service.omgevingswet.overheid.nl/publiek/omgevingsdocumenten/api/verbeelden/v3>
- Pre-productie omgeving:  
<https://service.pre.omgevingswet.overheid.nl/publiek/omgevingsdocumenten/api/verbeelden/v3>
- Ketenacceptatie omgeving:  
<https://service.acc.omgevingswet.overheid.nl/publiek/omgevingsdocumenten/api/verbeelden/v3>
- Ketenperformance omgeving:  
<https://service.perf.omgevingswet.overheid.nl/publiek/omgevingsdocumenten/api/verbeelden/v3>
- Integratieomgeving (kijkt ook naar de content van ketenacceptatie):  
<https://service.int.omgevingswet.overheid.nl/publiek/omgevingsdocumenten/api/verbeelden/v3>

De API specificaties zijn – net als de functionele documentatie – beschikbaar op het ontwikkelaarsportaal.

Om de APIs van de API-store te gebruiken is een API-key vereist. Deze is voor productie en pre-productie aan te vragen via:

<https://aandeslagmetdeomgevingswet.nl/ontwikkelaarsportaal/formulieren/api-key-aanvragen-0/>

## 4 Globale procesbeschrijving

### 4.1 Verbeelden API en PDOK Tile API in samenhang

Zoals in de inleiding is aangegeven verzorgt de Verbeelden API de objectverbeelding ('hoe' moet de visualisatie eruit zien) en de Tile API zorgt voor de geometrie ('waar' moet de visualisatie op de kaart staan). In deze paragraaf wordt eerst uitgebreid aangegeven welke processtappen er doorlopen moeten worden om met behulp van beide API's OW-objecten in een viewer te visualiseren. Het proces verloopt als volgt:

1. Op het endpoint `POST/symbolen/mapbox/_zoek`:
  - a. worden de te verbeelden OW-objecten in de vorm van zoekobjecten aangeleverd. Zoekobjecten bestaan minimaal uit een `primaireObjectIdentificator` en in de meeste gevallen bevatten ze ook een `secundaireObjectIdentificator`. In de OpenApiSpecificatie (OAS) is de lijst te vinden met mogelijkheden van de `primaireObjectIdentificator`. In hoofdstuk 6 is een lijst opgenomen met alle mogelijke combinaties van `primaireObjectIdentificatie` en `secundaireObjectIdentificatie`.<sup>10</sup>
  - b. wordt aangegeven via 'synchroniseerMetTileset' met welke tileset er gewerkt dient te worden
    - i. 'normaal' is voor de actueel geldige gegevens.
    - ii. 'volledig' is voor alle gegevens, ook historische en niet (meer) geldige gegevens.
  - c. Worden tijdreisparameters meegegeven (zie paragraaf 5.4).
2. In een layer van de Mapboxstyle (response van stap 1) zit een filter (zgn. hash) die zowel in de Verbeelden API als de PDOK Tile API terug te vinden is. De hash wordt via de GeoPackage aan PDOK aangeleverd (zie paragraaf 2.2 stap 2). Hiermee kan de verbeelding op de juiste versie van de locatie toegepast worden.
3. Voor het ophalen van de symbolen zijn er 4 endpoints voor het ophalen van de spritesheet, voor normale grootte verbeelding zijn dat `GET/spritesheet.json` en `GET/spritesheet.png`. Indien de symbolen 2 keer zo groot moeten zijn, dan zijn dat [GET/spritesheet@2x.json](#) en [GET/spritesheet@2x.png](#).
4. De geometrie en de verbeelding kunnen vervolgens tezamen in een viewer gevisualiseerd worden.
5. Eventueel kan er een legenda en label gemaakt worden met behulp van het resultaat van stap 1. De gegevens uit 'symboolmetadata' kunnen hiervoor gebruikt worden.

Eventueel kunnen stap 2 t/m 4 door een tool gedaan worden, bijvoorbeeld middels OpenLayers en de functie 'applyStyle' van de library [ol-mapbox-style](#). Deze functie haalt de spritesheet voor je op. De library checkt bij de tiles API over er een overeenkomst is tussen de hash die in de tile zit en die in het response van stap 1 (response Verbeelden API) zit.

---

<sup>10</sup> Voor stap 1 is het een voorwaarde dat de identificatie van het CIM-OW object bekend is. Deze kan bijvoorbeeld uit de Omgevingsdocumenten Presenteren API komen.

## 4.2 Genereren vectortiles

Het proces van het ontvangen van locaties (aangeleverd via omgevingsdocumenten) tot en met het ontsluiten van tiles verloopt in 3 stappen:

1. Ozon maakt een GeoPackage op 2 momenten:
  - I. Ozon ontvangt via het registratiekoppelvlak nieuwe locaties. Meestal in een batch rond 9:00, tenzij er geen plannen zijn aangeleverd aan de planketen. Soms ontvangt Ozon de locaties op andere tijdstippen gedurende de dag. Een half uur na iedere registratie, mits er niet nog een registratie in dat half uur volgt, zal er een GeoPackage gemaakt worden van de actuele locaties en naar de deliveryservice van PDOK gestuurd worden. Wanneer PDOK op dat moment nog bezig is met het genereren van vectortiles, dan zal de opdracht tot het maken (en versturen) van een GeoPackage uitgesteld worden tot de nog lopende opdracht afgerond is.
  - II. Ozon maakt dagelijks (rond middernacht) één GeoPackage – van alle versies van de locaties in de database – en levert deze aan op de deliveryservice van PDOK. Het verschil met I is dat in deze GeoPackage alle versies van de locaties zitten, dus ook versies die niet meer juridisch geldig zijn.
2. PDOK genereert vectortiles op basis van de aangeleverde GeoPackage (dat duurt over het algemeen ongeveer 2 uur).
3. Na succesvolle verwerking wordt de tileset ontsloten via [https://service.pdok.nl/omgevingswet/{DATASET}/wmts/v1\\_0/WMTSCapabilities.xml](https://service.pdok.nl/omgevingswet/{DATASET}/wmts/v1_0/WMTSCapabilities.xml), zie bijv. [omgevingsdocumenten-demo](#) (zie paragraaf 4.2 voor andere omgevingen).

Vanaf stap 4 is het mogelijk om via een gebruikerstoepassing (zoals één van de DSO-viewers) de vectortiles van een locatie op verschillende schaalniveaus met een verbeelding te zien.



### 4.3 Aandachtspunten bij genereren vectortiles

Ten behoeve van goede zichtbaarheid in een viewer op verschillende schaalniveaus worden de lijn- en vlak geometrieën versimpeld. Punt geometrieën worden niet versimpeld. In de rest van deze paragraaf wordt toegelicht wat er tijdens het genereren van de vectortiles gedaan wordt, om een goede zichtbaarheid in een viewer te realiseren.

Enkele aandachtspunten die relevant zijn voor de gegenereerde vector tiles:

1. Op dit moment worden alleen vector tiles gegenereerd volgens de RD tilematrixset conform de [Nederlandse richtlijn tiling](#) (Bijlage B: Well-known scale set Nederland)<sup>11</sup>.
2. Op dit moment worden vector tiles gegenereerd in de RD tilematrixset van zoomlevel 0 tot zoomlevel 11. Zie Tabel 1 voor specificaties van de verschillende zoomniveaus.
3. Op dit moment wordt er een simplificatie uitgevoerd op de zoomniveau's 0 t/m 10. Het gaat hierbij om de volgende 3 operaties, geparametriseerd per zoomlevel:
  - a) Polygonen worden gezeefd per zoomniveau. Het zeven is nodig, omdat het anders resulteert in ongeldige geometrieën in de resulterende vector tiles. Voor zoomniveau  $z$  geldt dat geometrieën die kleiner zijn qua oppervlakte dan  $resolution_z * resolution_z$  eruit worden gezeefd. In het geval van gebiedslocaties kan het dus voorkomen dat deze in een vectortile niet zichtbaar wordt bij een bepaald schaalniveau op de kaart. Deze locatie bestaat dan dus wel in het stelsel, maar wordt op die schaal niet weergegeven. Onderstaande afbeelding laat dit effect zien voor de gebieden binnen de paarse cirkel die op zoomniveau 8 nog wel zichtbaar zijn maar niet meer op zoomniveau 4.



<sup>11</sup> De richtlijn laat het na om deze tilematrixset een eenduidige identifier te geven, in de praktijk wordt "EPSG:28992" als identifier gebruikt. Zie ook GH [issue](#) in praktijkrichtlijn-vector-tiling.

- b) Op de lagen 0 t/m 10 van de GeoPackage worden geometrieën versimpeld (dus zowel POLYGON, LINESTRING als POINT, maar dit is op POINT-lagen niet van toepassing). De simplificatie is nodig, omdat het anders resulteert in ongeldige geometrieën in de resulterende vector tiles, wat weer leidt tot rendering artifacts in het kaartbeeld. De waarde die gebruikt wordt voor de simplify tolerance is  $resolution\_z/4$ . In het geval van zoomlevel 9 is de simplify tolerance dus  $6,72/4=1,68$ , zie ook de tabel.

Tabel 1: Specificaties van vector tiles zoom niveaus

Zoomlevel	Resolutie (m)	Simplificatie	3b) Simplificatie tolerantie (m)	Zeef grenswaarde (m)
0	3440,640	ja	860,16	3440,640
1	1720,320	ja	430,080	1720,320
2	860,160	ja	215,040	860,160
3	430,080	ja	107,520	430,080
4	215,040	ja	53,760	215,040
5	107,520	ja	26,880	107,520
6	53,760	ja	13,440	53,760
7	26,880	ja	6,720	26,880
8	13,440	ja	3,360	13,440
9	6,720	ja	1,680	6,720
10	3,360	ja	0,84	3,360
11	1,680	nee	n.v.t.	1,680

4. De simplificatie wordt **niet** toegepast op het diepste zoomlevel (te weten 11). Dit zorgt ervoor dat bij overzoomen op de vector tiles (dat wil zeggen de tegels van z11 bekijken op z12 of dieper) simplificaties niet direct zichtbaar zijn. Echter als ver genoeg ingezoomd wordt worden de simplificaties die inherent zijn aan het vector tile formaat zichtbaar. Een vector tile zelf bestaat namelijk uit een discreet raster van 4096 bij 4096, waarop de geometrieën geëncodeerd worden, zie ook de [Mapbox Vector Tile Specification](#) voor meer informatie.

## 4.4 Omgevingsdocumenten Tiles API (WMTS)

De PDOK Tiles API is in verschillende omgevingen te bereiken via PDOK. Er zijn aparte endpoints voor de actuele set (URL zonder '-totaal') en volledige set (URL met '-totaal'). In de laatste set is het mogelijk om vectortiles uit het verleden en de toekomst op te halen. :

- Demo omgeving:  
[https://service.pdok.nl/omgevingswet/omgevingsdocumenten-demo/wmts/v1\\_0](https://service.pdok.nl/omgevingswet/omgevingsdocumenten-demo/wmts/v1_0)  
[https://service.pdok.nl/omgevingswet/omgevingsdocumenten-totaal-demo/wmts/v1\\_0](https://service.pdok.nl/omgevingswet/omgevingsdocumenten-totaal-demo/wmts/v1_0)
- Productie omgeving:  
[https://service.pdok.nl/omgevingswet/omgevingsdocumenten/wmts/v1\\_0](https://service.pdok.nl/omgevingswet/omgevingsdocumenten/wmts/v1_0)  
[https://service.pdok.nl/omgevingswet/omgevingsdocumenten-totaal/wmts/v1\\_0](https://service.pdok.nl/omgevingswet/omgevingsdocumenten-totaal/wmts/v1_0)
- Pre-productie omgeving:  
[https://service.pdok.nl/omgevingswet/omgevingsdocumenten-pre/wmts/v1\\_0](https://service.pdok.nl/omgevingswet/omgevingsdocumenten-pre/wmts/v1_0)  
[https://service.pdok.nl/omgevingswet/omgevingsdocumenten-totaal-pre/wmts/v1\\_0](https://service.pdok.nl/omgevingswet/omgevingsdocumenten-totaal-pre/wmts/v1_0)
- Ketenacceptatie omgeving:  
[https://service.pdok.nl/omgevingswet/omgevingsdocumenten-ketenacc/wmts/v1\\_0](https://service.pdok.nl/omgevingswet/omgevingsdocumenten-ketenacc/wmts/v1_0)  
[https://service.pdok.nl/omgevingswet/omgevingsdocumenten-totaal-ketenacc/wmts/v1\\_0](https://service.pdok.nl/omgevingswet/omgevingsdocumenten-totaal-ketenacc/wmts/v1_0)
- Ketenperformance omgeving:  
[https://service.pdok.nl/omgevingswet/omgevingsdocumenten-ketenacc/wmts/v1\\_0](https://service.pdok.nl/omgevingswet/omgevingsdocumenten-ketenacc/wmts/v1_0)  
[https://service.pdok.nl/omgevingswet/omgevingsdocumenten-totaal-ketenacc/wmts/v1\\_0](https://service.pdok.nl/omgevingswet/omgevingsdocumenten-totaal-ketenacc/wmts/v1_0)

De PDOK Tiles API-specificatie is te vinden via het end-point via WMTSCapabilities.xml, bijvoorbeeld: [https://service.pdok.nl/omgevingswet/omgevingsdocumenten-demo/wmts/v1\\_0/WMTSCapabilities.xml](https://service.pdok.nl/omgevingswet/omgevingsdocumenten-demo/wmts/v1_0/WMTSCapabilities.xml).

## 5 Endpoints en schema's

De endpoints van de Verbeelden API zijn ontworpen om de OW-objecten zo uit te leveren dat ze gevisualiseerd kunnen worden op een kaart. De Verbeelden API maakt geen gebruik van paginering. In de API specificatie van de Verbeelden API is een lijst van alle operaties en resources te vinden, en daarnaast een beschrijving van alle gebruikte objecten (schema's), zowel voor uitlevering als voor bevraging van de Verbeelden API.

### 5.1 /symbolen/mapbox/\_zoek

Van CIM-OW-objecten zoals Activiteiten, Gebiedsaanwijzingen, Omgevingsnormen, Omgevingswaarden en Kaarten worden de symboolcodes met behulp van de groep die bij het betreffende Object hoort uit de waardelijsten gehaald en uitgeleverd in de mapboxstyle. Meer informatie over mapbox: [Mapbox Vector Tile Specification](#)<sup>12</sup>. Bij afwijking van de standaardsymbolisatie wordt de symboolcode uit het aangeleverde SymbolisatieItem gehaald.

Het opvragen van de mapboxstyle kan door middel van het aanleveren van een zoekobject, zoals beschreven in paragraaf 2.1 stap 1.

Een locatie met eenzelfde identificatie kan bij meerdere andere objecttypen horen zoals een Omgevingsnorm en een Activiteit. Je kan de Verbeelden API dus alleen bevragen met een zoekobject specifiek voor het objecttype. Met andere woorden, er kunnen meerdere verbeeldingen bij een locatie horen. Om te weten welke nodig is moeten de verschillende endpoints van de Verbeelden API bevroegd worden.

### 5.2 /symbolen

Binnen deze API zijn er nog twee andere endpoints om de symbolisering op te halen.

- /symbolen/findByType/{symbooltype}

Via dit endpoint kan alle symbolisatie voor een bepaald symbooltype worden opgehaald. Mogelijke waarden die ingevuld kunnen worden zijn *punt*, *lijn*, *vlak* en *normwaarde*. Dit resulteert in een lijst met symbolen met hun karakteristieke eigenschappen. Bij bepaalde symbolen kan het voorkomen dat er een afbeelding wordt gebruikt. De afbeeldingen komen van de spritesheet, zie paragraaf 5.3.

- /symbolen/findById/{symboold}

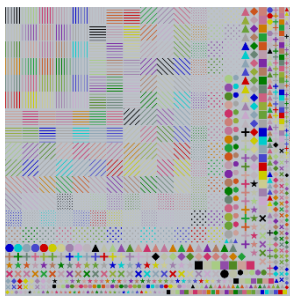
Via dit endpoint kan de symbolisatie-informatie van één symboold worden opgevraagd.

---

<sup>12</sup> <https://docs.mapbox.com/vector-tiles/specification/>

### 5.3 /spritesheet

De afbeeldingen van de symbolen worden geleverd in één grote afbeelding; hiervoor wordt CSS Sprite gebruikt. Hiermee worden de CIM-OW-objecten gevisualiseerd. Er zijn vier endpoints die spritesheet-informatie teruggeven. Endpoint `/spritesheet.png` geeft onderstaand plaatje terug waarin alle standaardsymbolisatie die plaatjes gebruiken is te vinden. Via `/spritesheet.json` kan aan de hand van de code van het benodigde plaatje informatie worden gevonden waar dit subplaatje zich in `spritesheet.png` bevindt.



(voorbeeld `spritesheet.png`)

```
1 {  
2   "pc002.png": {  
3     "x": 637,  
4     "y": 196,  
5     "width": 24,  
6     "height": 24  
7   },  
8   "pc003.png": {  
9     "x": 637,  
10    "y": 221,  
11    "width": 24,  
12    "height": 24  
13  },  
14  "pc004.png": {  
15    "x": 637,  
16    "y": 246,  
17    "width": 24,  
18    "height": 24  
19  }  
20  ...  
21 }
```

(voorbeeld `spritesheet.json`)

Daarnaast zijn er ook spritesheet-endpoints met een tweemaal zo grote resolutie, via `/spritesheet@2x.json` en `/spritesheet@2x.png`.

## 5.4 Tijdreizen

De Verbeelden API biedt features voor tijdreizen. Hiermee wordt het mogelijk om regels uit het verleden inzien, maar geen regels van vóór de datum van inwerking treden van de Omgevingswet. Het is zowel mogelijk om te tijdreizen langs de geldigheidsas (geldig+in werking), als langs de beschikbaarheidsas. In het eerste geval wordt de vraag gesteld: wat was of wordt er geldig? In het tweede geval wordt de vraag gesteld: wat had ik destijds gezien als ik deze API had bevestigd op het gegeven tijdstip? De POST/symbolen/mapbox/\_zoek van de Verbeelden API biedt drie parameters voor tijdreizen:

- a) `geldigOp`: optioneel. Het is logisch deze in te vullen bij tijdreizen en dan ook te synchroniseren met de volledige tileset.
- b) `inWerkingOp`: optioneel. Het is logisch deze in te vullen bij tijdreizen en dan ook te synchroniseren met de volledige tileset
- c) `beschikbaarOp`: optioneel. Wanneer je wel de parameter 'synchroniseerMetTileset' invult en niet de parameter `beschikbaarOp`, dan wordt het tijdstip toegepast waarop de huidige tileset oorspronkelijk werd aangeboden aan PDOK. Dit wordt ook gedaan op alle plekken in de presenteren API v7. Dit is een impliciete tijdreisvraag, die correspondeert met de vraag 'op welke informatie in de API is de tileset gebaseerd?' Hiermee wordt voorkomen dat data getoond wordt, vóórdat de vectortiles hiervan beschikbaar zijn.

Tot slot is ook het tijdstip op te halen waarop de geopackage voor de vectortiles als laatste is aangeboden. Dit gaat via een GET op het endpoint `/tiling`. Dit tijdstip is hetzelfde als het tijdstip dat gebruikt wordt door de parameter 'synchroniseerMetTileset'. Dit tijdstip geeft de `beschikbaarOp`-tijdreisparameter aan naar de informatie in de Presenteren en Verbeelden API's, die correspondeert met de momenteel beschikbare actuele tileset.

## 5.5 /health en /app-info

De Verbeelden API heeft een `/health` endpoint en een `/app-info` endpoint, waarop informatie over de status van de API, respectievelijk de momenteel van toepassing zijnde informatiemodelversies te vinden zijn.

## 6 Zoekobjecten

### Vastgestelde zoekobjecten

**1. GebiedsaanwijzingZoekobject**

*primairObjectIdentificatie*: nl.imow-[bg-code].gebiedsaanwijzing.[x]  
*secundaireObjectIdentificatoren*: n.v.t.

**2. OmgevingsnormNormwaardeZoekobject**

*primairObjectIdentificatie*: nl.imow-[bg-code].omgevingsnorm.[x]  
*secundaireObjectIdentificatoren*: nl.imow-[bg-code].normwaarde.[x]

**3. OmgevingswaardeNormwaardeZoekobject**

*primairObjectIdentificatie*: nl.imow-[bg-code].omgevingswaarde.[x]  
*secundaireObjectIdentificatoren*: nl.imow-[bg-code].normwaarde.[x]

**4. RegeltekstActiviteitlocatieaanduidingZoekobject**

*primairObjectIdentificatie*: nl.imow-[bg-code].regeltekst.[x]  
*secundaireObjectIdentificatoren*: nl.imow-[bg-code].activiteitlocatieaanduiding.[x]

**5. KaartKaartlaagZoekobject**

*primairObjectIdentificatie*: nl.imow-[bg-code].kaart.[x] of  
nl.imow-[x].kaartlaag.[x]  
*secundaireObjectIdentificatoren*: nl.imow-[bg-code].kaartlaag.[x]

**6. WerkingsgebiedZoekobject**

*primairObjectIdentificatie*: nl.imow-[bg-code].regeltekst.[x]  
, nl.imow-[bg-code].divisie.[x] en nl.imow-[bg-code].divisietekst.[x]  
*secundaireObjectIdentificatoren*: optioneel, maar wordt wel aangeraden.  
! Let op: als je zonder secundair gebruikt, krijg je alle locaties die bij de regeltekst, divisie of divisietekst horen.

**7. RegelingsgebiedZoekobject**

*primairObjectIdentificatie*: AKN van het omgevingsdocument  
*secundaireObjectIdentificatoren*: optioneel  
nl.imow-[bg-code].regelingsgebied.[x]

## Ontwerp zoekobjecten

### 8. **OntwerpGebiedsaanwijzingZoekobject**

primairObjectTechnischId: gelijk aan vastgesteld, maar dan technischID.  
secundaireObjectIdentificatoren: n.v.t.

### 9. **OntwerpOmgevingsnormNormwaardeZoekobject**

primairObjectTechnischId: gelijk aan vastgesteld, maar dan technischID.  
secundaireObjectIdentificatoren:  
gelijk aan vastgesteld, maar dan technischID.

### 10. **OntwerpOmgevingswaardeNormwaardeZoekobject**

primairObjectTechnischId: gelijk aan vastgesteld, maar dan technischID.  
secundaireObjectIdentificatoren:  
gelijk aan vastgesteld, maar dan technischID.

### 11. **OntwerpRegeltekstActiviteitlocatieaanduidingZoekobject**

primairObjectTechnischId: gelijk aan vastgesteld, maar dan technischID.  
secundaireObjectIdentificatoren:  
gelijk aan vastgesteld, maar dan technischID.

### 12. **OntwerpKaartKaartlaagZoekobject**

primairObjectTechnischId: gelijk aan vastgesteld, maar dan technischID.  
secundaireObjectIdentificatoren: gelijk aan vastgesteld, maar dan technischID.

### 13. **OntwerpWerkingsgebiedZoekobject**

primairObjectIdentificatie: gelijk aan vastgesteld, maar dan technischID.

secundaireObjectIdentificatoren: optioneel, maar wordt wel aangeraden. Dit id kan zowel ontwerp (locatie.technisch-id) als vastgestelde (locatie.id) zijn.

! Let op: als je zonder secundair gebruikt, krijg je alle locaties die bij de (ontwerp)regeltekst, (ontwerp)divisie of (ontwerp)divsietekst horen.

### 14. **OntwerpRegelingsgebiedZoekobject**

primairObjectTechnischId: gelijk aan vastgesteld, maar dan technischID.  
secundaireObjectIdentificatoren: gelijk aan vastgesteld, maar dan technischID.