

# Data Op Kaart / Export Inrichtingen als WFS

## Doel/toepassing

Het regelmatig aanleveren van gestructureerde informatie uit de OpenWave inrichtingstabellen aan een extern GEO-systeem. De data worden als WFS-file (geojson) gescheduled vanuit OpenWave geëxporteerd naar een plek waar het GEO-systeem van de gebruiker bij kan. Met de geleverde data kan het GEO-systeem (indien daartoe geprepareerd) in hun dataportaal een set inrichtingen als punten op de kaart tonen die door de gebruiker op de kaart live door middel van trechtering op thema's/subthema's wordt samengesteld uit het domein van alle inrichtingen uit de OpenWave database.

Elke feature in de WFS-file is een inrichting met een aantal vaste properties: de punt-geometrie, de inrichtingsnaam, een hyperlink naar het inrichtingsportaal in OpenWave, en het adres en de plaats van de vestiging. Alle inrichtingen zijn in de WFS-file opgenomen. Daarnaast zijn er één of meer flexibele properties (dat kunnen er makkelijk 100 zijn) die bepaald worden door de vulling van de thema-tabellen en de SQL-propertytabel (zie hieronder).

Die flexibele properties zijn gegroepeerd per thema/subthema. Thema's zijn bijvoorbeeld duurzaamheid, gemeente en tanks. Subthema's bij duurzaamheid zijn bijvoorbeeld energieverbruik of vervoer-transportbewegingen. De property zelf binnen energieverbruik is dan bijvoorbeeld elektraverbruik kleiner of gelijk aan 50.000 kWh of elektraverbruik groter dan 50.000 kWh. De properties binnen een thema/subthema - mits goed gedefinieerd - sluiten elkaar uit. Een inrichting kan maar in één property per thema/subthema vallen.

Voorbeeld van één featuretype (= inrichting) binnen de WFS-file met twee verschillende flexibele properties:

```
{
  "type": "FeatureCollection",
  "totalFeatures": 2,
  "features": [
    {
      "type": "Feature",
      "id": "OpenWave_Inrichting_11420",
      "geometry": {
        "type": "Point",
        "coordinates": [
          176120,
          446171
        ]
      },
      "geometry_name": "Point",
      "properties": {
        "inrichtingnaam": "Pauls eerste inrichting",
        "objadres": "Prins Hendrikweg 3017",
        "objplaats": "Bennekom",
        "url": "https://demo2.open-wave.nl/#inrichtingdetail/11420",
        "inrichtingsindelingen_amvb-types": "Vergunningsplichtig",
        "toezicht_bezoeken_nooitbezoekt": "Ja"
      }
    },
  ],
}
```

Thema en subthema zijn in de WFS-file onderdeel van de (flexibele) propertynaam. De drie entiteiten zijn hierin gescheiden door een underscore. Thema is verplicht, subthema is niet verplicht. In bovenstaand voorbeeld is de property *amvb-types* alleen gekoppeld aan het thema *inrichtingsindelingen*. De property *nooitbezoekt* is ingedeeld onder het subthema *bezoeken* binnen het thema *toezicht*.

Met bovenstaande structurering en aanlevering van de WFS-file zou de gebruiker van het externe GEO-systeem (indien dit systeem de WFS-file als zodanig accepteert en bewerkt) live op de kaart vragen kunnen stellen als: geef mij de inrichtingen waarvoor geldt dat ze in gemeente x liggen, en dat daar de afgelopen jaar geen toezicht is geweest, en dat een elektraverbruik groter dan 50.000 KWh is gerapporteerd, en dat er meer dan 200 koeien zijn. Het bevragen gebeurt dan door het aanklikken van thema's, subthema's en propertywaarden.

## Definitie properties, thema's en subthema's

In het *inrichtingenbeheerportaal* bestaat onder de kolom *Overig* de tegel *thema/subthema export inrichtingen*. Hierachter kan een indeling gemaakt worden op thema en subthema. Een thema kan, maar hoeft niet, ingedeeld te worden in subthema's. Een (sub)thema naam mag alleen bestaan uit letters of cijfers of een hyphen (-). **Dus geen spaties of underscores**. De tabelnamen zijn *tbexportinrkt\_thema* en *tbexportinrkt\_subthema*.

Ook in het *inrichtingenbeheerportaal* onder de kolom *Overig* de tegel *SQL export inrichtingen*. Hierin worden de propertynamen gedefinieerd en gekoppeld aan een thema/subthema. Bij de propertynaam, die ook alleen mag bestaan uit letters, cijfers of een hyphen, hoort een SQL/statement waarmee de mogelijke waarde van de property per inrichting wordt bepaald. De tabelnaam is

tbexportinrkt\_sql.

Een voorbeeld van een indeling:

### SQL-lijst export inrichtingen

ID	Thema	SubThema	Property
2	duurzaamheid	energieverbruik	elektra
73	duurzaamheid	energieverbruik	TEST(1)
3	duurzaamheid	energieverbruik	gas
32	duurzaamheid	vervoer-transportbewegingen	meerdan500
30	duurzaamheid	vervoer-transportbewegingen	tussen1en100
31	duurzaamheid	vervoer-transportbewegingen	tussen100en500
4	inrichtingsindelingen		sbi
5	inrichtingsindelingen		amvb-types
6	inrichtingsindelingen		gemeentes
37	toezicht	geplande-trajecten	komendjaar
38	toezicht	geplande-trajecten	over2jaarofverder
39	toezicht	geplande-trajecten	tussen1en2jaar
36	toezicht	geplande-trajecten	geen
35	toezicht	overtredingen	minstens1afgelopen5jaar
33	toezicht	overtredingen	geenafgelopen5jaar

Achter elke gedefinieerde property is een SQL-statement gedefinieerd dat bij de export naar de WFS-file voor elke inrichting wordt geëvalueerd. Elke inrichting krijgt bij elke property bij de evaluatie van de SQL dus een propertywaarde toegekend. Alle mogelijke waarden per property bepalen in de externe Geoserver de trechtermogelijkheden.

De resultaatset van elk SQL-statement moet bestaan uit twee kolommen: de inrichtingskey (tbmilinrichtingen.dnkey) gevolgd door de waarde. Tijdens het maken van de WFS-file zullen alle queries voor alle inrichtingen worden doorlopen (althans voor de niet-vervallen inrichtingen die gekoppeld zijn aan een perceeladres met gevulde x- en y-coördinaat). In de loop zal de variabele {id} in de query worden vervangen door de dnkey van de actieve inrichting uit de loop.

Twee voorbeelden:

Thema: toezicht, subthema: geplande inspecties en property: over2jaarofverder

```
select c.dnkey dnkeymilinrichtingen,
       case when exists
           (select 1 from tbinspecties a
            where a.dnkeymilinrichtingen = c.dnkey
            and a.ddrappel > fn_vandaag(730) and a.ddcontrole is null)
           then 'ja'
       else 'nee' end janee
```

```
from tbmilinrichtingen c
    inner join tbperceeladressen d
        on (c.dnkeyperceeladressen = d.dnkey
            and d.dnxcoordinaat is not null and d.dnycoordinaat
is not null)
where c.ddbedrijfeinde is null
and c.dnkey = {id}
```

Bovenstaand voorbeeld levert dus per inrichting een resultaatset op met de propertywaarde ja of nee (er is wel een traject gepland over 2 jaar of niet). Tijdens de trechtering kan bij dit thema alleen ja of nee worden aangekruist.

Thema: inrichtingsindelingen, subthema: null en property: gemeentes

```
select a.dnkey dnkeymilinrichtingen, b.dvgemeentenaam
from tbmilinrichtingen a
    inner join vwfrmlokaties b
        on (a.dnkeyperceeladressen = b.dnkeyperceeladressen
            and b.dnxcoordinaat is not null and b.dnycoordinaat is
not null)
where a.ddbedrijfeinde is null
and a.dnkey = {id}
```

Bovenstaand voorbeeld levert dus per inrichting een resultaatset op met als propertywaarde de gemeentenaam waar de inrichting zich bevindt. Tijdens de trechtering kan bij dit thema gekozen worden uit alle gemeentenamen die bij de inrichtingen voorkomen.

## Aanroep van de export via Taakscheduler

Door een kaart in de tabel `tbtaskscheduler` op te nemen (portaal *Service centrum*, kolom *Acties*) kan de samenstelling van de WFS-export en verzending daarvan gescheduled worden gestart. De aanroep (de taak) is `exportInrichtingenWFS` (zie: [Taakscheduler](#)). Het samenstellen van de file kan geruime tijd duren: voor elke inrichting worden evenzoveel SQL's geëvalueerd als er opgenomen zijn in de tabel `tbexportinrkr_tsql` (40 SQL's voor 10.000 inrichtingen duurt ongeveer een uur).

Als de export (runnable) begint wordt de Datum van de instelling *Sectie: Operations, Item: exportInrichtingenWFS* gevuld met timestamp. De kolom *Tekst* met medewerkerscode en *Getail* met 1. Indien klaar dan wordt *Getail* op null gezet. Ook in de operationslog wordt onder de code `exportInrichtingenWFS` de voortgang bijgehouden.

De WFS-file onder de naam `OWEXportInrkrWFS.Json` wordt geplaatst op de map gedefinieerd door de kolom *Tekst* van de instelling *Sectie: Inrichtingen* en *Item: EndpointExportInrkrWFS* op de server waar ook OpenWave is geïnstalleerd. Dit is een noodzakelijke instelling. De file wordt elke keer overschreven. Aangezien de map waarop de file geplaatst wordt benaderbaar moet zijn door het externe GEO-systeem zal deze in samenspraak met REM worden afgesproken. Waarschijnlijk op een submap van de geïnstalleerde Geoserver.

Het verwerken van alle inrichtingen kan erg lang kan duren, wat het testen van de gemaakte rapporten lastig maakt. Daarom is het mogelijk om de taskscheduler een testrapport met een aangepaste hoeveelheid inrichtingen te laten maken. Dit kan met de instelling *Sectie: Operations* en *Item: ExportWFSTestRapport*. Het default aantal inrichtingen dat verwerkt wordt met deze instelling aangevinkt is 10. Met *Getal1* kan deze hoeveelheid aangepast worden.

## Geo/Kaart

From:

<https://doc.open-wave.nl/> - Documentatie

Permanent link:

[https://doc.open-wave.nl/doku.php/openwave/1.34/applicatiebeheer/instellen\\_inrichten/data\\_op\\_kaart?rev=1766058314](https://doc.open-wave.nl/doku.php/openwave/1.34/applicatiebeheer/instellen_inrichten/data_op_kaart?rev=1766058314)

Last update: **2025/12/18 12:45**

