CRS extensions for spatial APIs

Public session 1: 2021-11-04

Overview

- Introduction
- Goal
- Narrative
- Demo
- Output formats

Introduction – GIS Specialisten

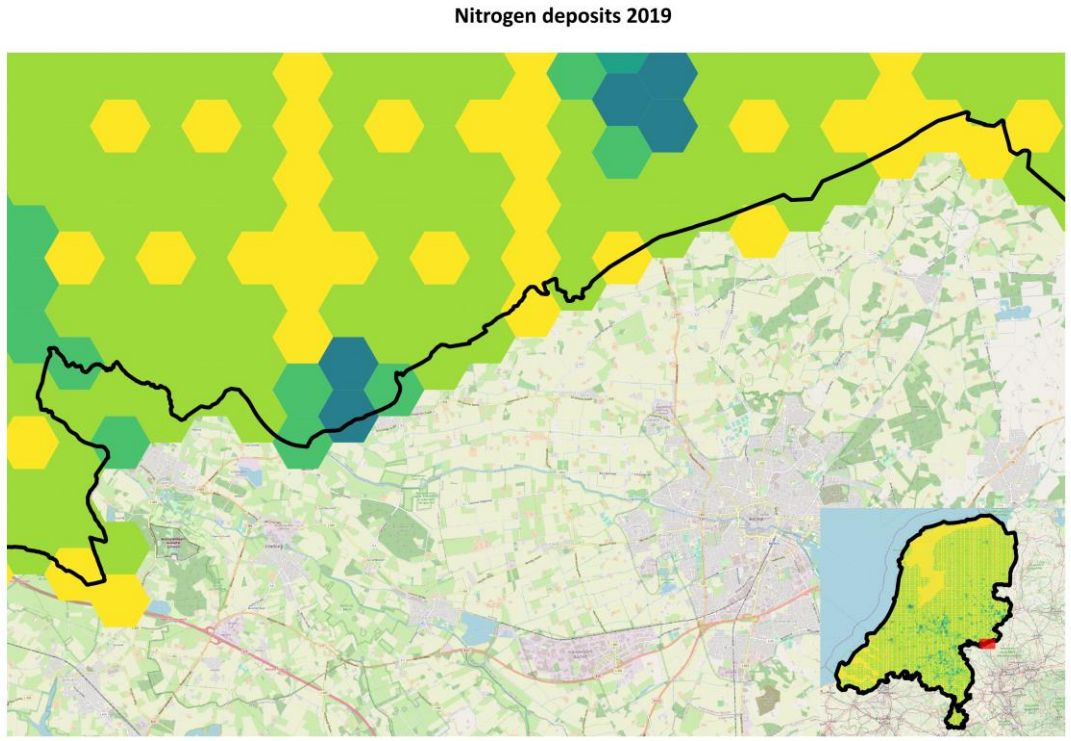
- Geo-ICT with 13 years of experience
- 20+ colleagues from 7 different countries
- Located in Utrecht
- Geo-ICT services:
    - Creation of tailor made Spatial Data Infrastructures
    - Operation of SDI's
    - GIS analysis
- Product development: Geoservice.Cloud
    - Gis-as-a-Service
    - Hosting, services, analyses and online presentation in Web-GIS
    - Build on Open Source, runs in the cloud
- Working through co-creation

Goal

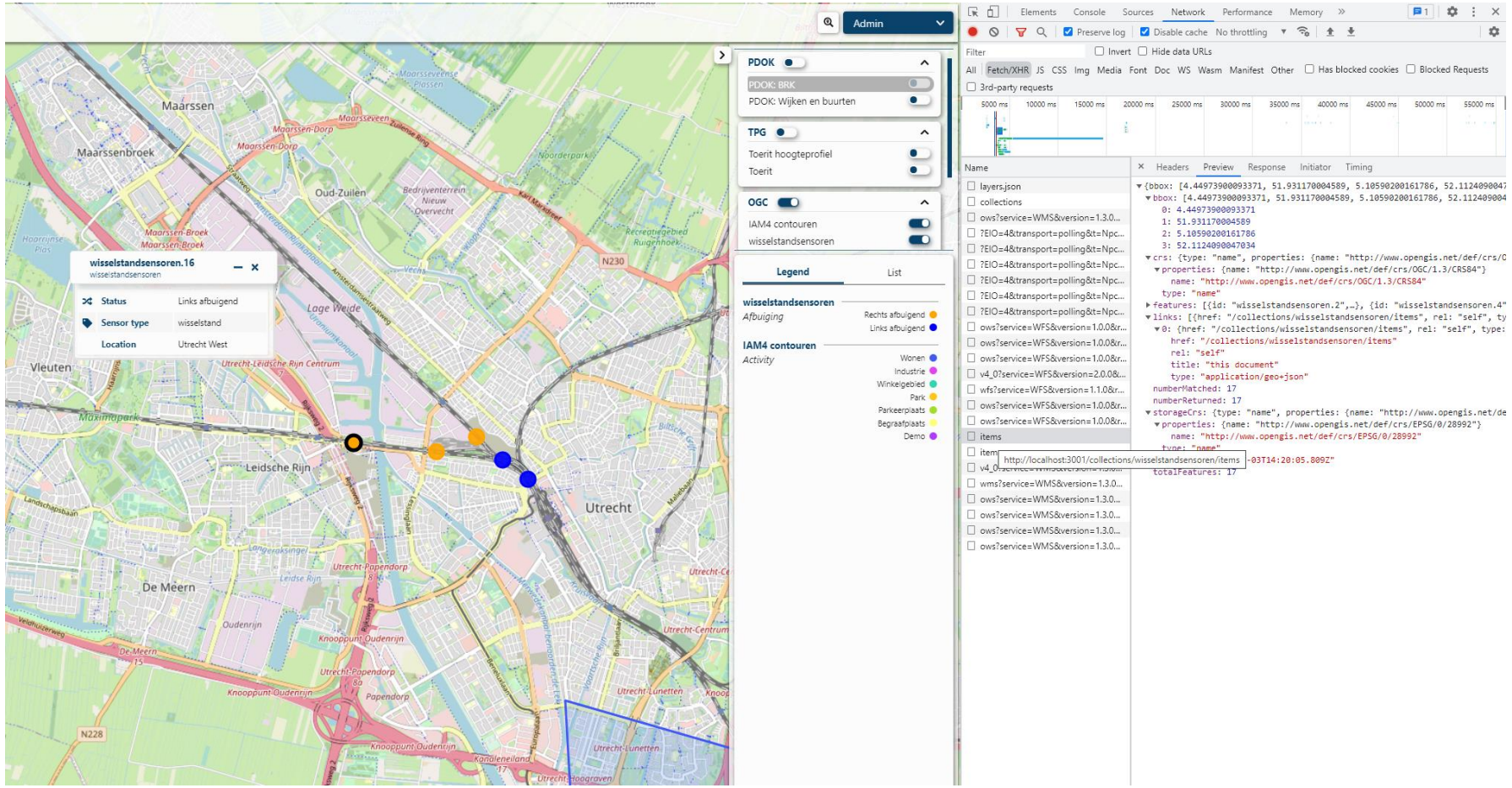
1. Demonstrate how one can serve spatial data in both RD and ETRS89 CRS in such a way that the implementation is OGC API Features – Part 2: CRS-compliant.
2. Demonstrate how the effort, needed to serve spatial data in both RD and ETRS in accordance with OGC API Features – Part 2: CRS, relates to the effort, needed to serve the same spatial data in those CRSs in accordance with the Dutch API strategy (using content negotiation for CRS).
3. Demonstrate how easy or difficult it is to adapt an existing API that uses content negotiation for CRS, to a version that follows the OGC API Features – Part 2: CRS specification, or to a version that supports both mechanisms.
4. Demonstrate, by using at least one client, how users can request data in a specific CRS from a.) an API that implements the OGC API Features – Part 2: CRS specification and b.) an API that implements the Dutch API Strategy mechanism with content negotiation for CRS.
5. Demonstrate how your favorite (i.e. useful / user-friendly / original / popular) existing API that supports multiple CRSs, can be converted into a version that is OGC API Features – Part 2: CRS-compliant.

Narrative

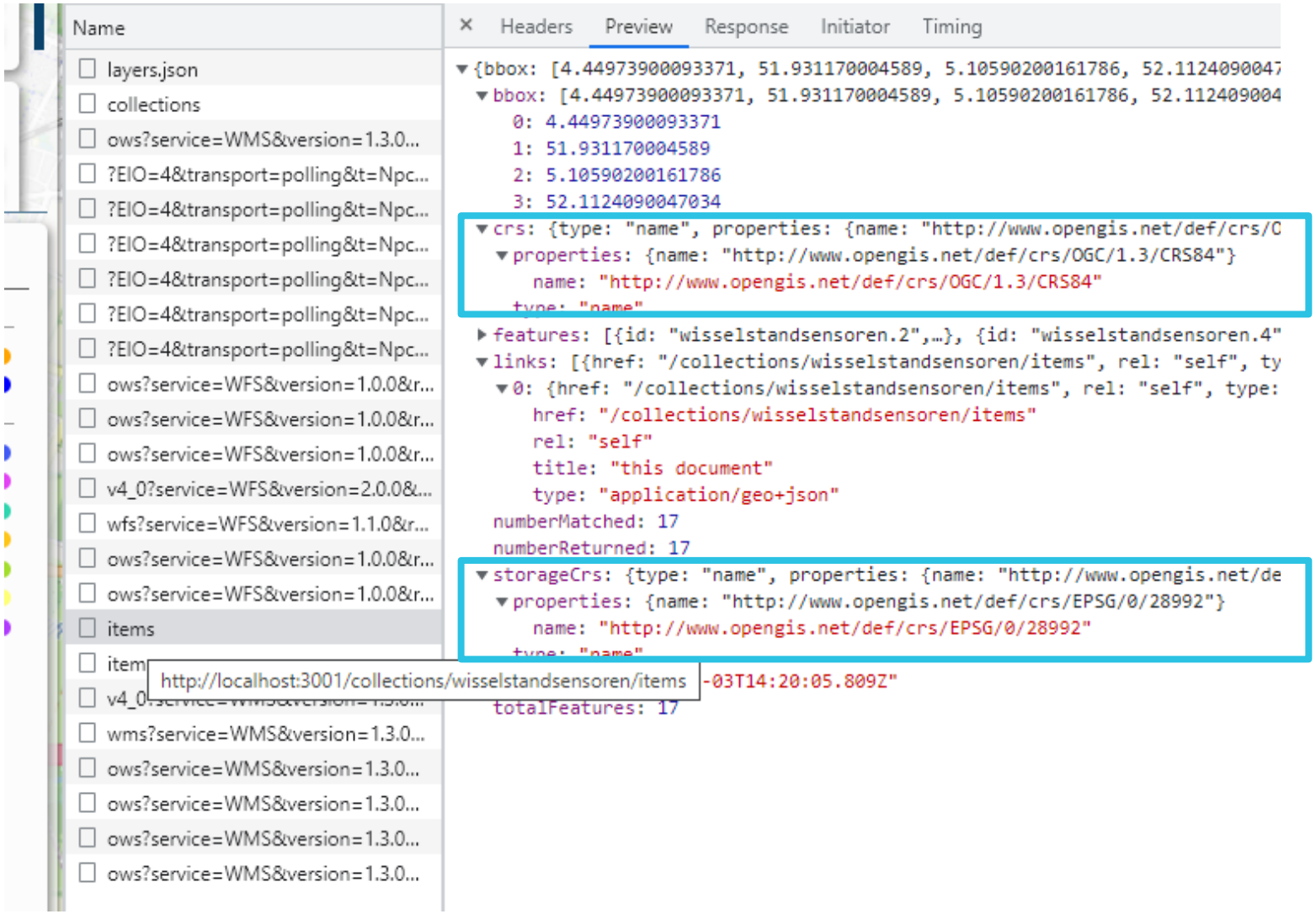
- INSPIRE Nitrogen deposits
- National dataset that is also important to neighbouring countries
    - Need for CRS extension



Demo


Admin
PDOK
PDOK: BRK
PDOK: Wijken en buurten
TPG
Toerit hoogteprofiel
Toerit
OGC
IAM4 contouren
wisselstandsensoren
Legend
List
wisselstandsensoren
Afbuiging
Rechts afbuigend
Links afbuigend
IAM4 contouren
Activity
Wonen
Industrie
Winkelgebied
Park
Parkeerplaats
Begraafplaats
Demo
wisselstandsensoren.16
wisselstandsensoren
Status
Links afbuigend
Sensor type
wisselstand
Location
Utrecht West
Maarssen
Maarssenbroek
Oud-Zuilen
Vleuten
De Meern
Leidsche Rijn
Utrecht
Lage Weide
Maximapark
N230
N228
Elements
Console
Sources
Network
Performance
Memory
Preserve log
Disable cache
No throttling
Filter
Invert
Hide data URLs
Fetch/XHR
Has blocked cookies
Blocked Requests
3rd-party requests
Name
Headers
Preview
Response
Initiator
Timing
layers.json
collections
items
totalFeatures: 17

Demo


Name
layers.json
collections
ows?service=WMS&version=1.3.0...
?EIO=4&transport=polling&t=Npc...
?EIO=4&transport=polling&t=Npc...
?EIO=4&transport=polling&t=Npc...
?EIO=4&transport=polling&t=Npc...
?EIO=4&transport=polling&t=Npc...
?EIO=4&transport=polling&t=Npc...
ows?service=WFS&version=1.0.0&r...
ows?service=WFS&version=1.0.0&r...
ows?service=WFS&version=1.0.0&r...
v4_0?service=WFS&version=2.0.0&...
wfs?service=WFS&version=1.1.0&r...
ows?service=WFS&version=1.0.0&r...
ows?service=WFS&version=1.0.0&r...
items
item
v4_0
wms?service=WMS&version=1.3.0...
ows?service=WMS&version=1.3.0...
ows?service=WMS&version=1.3.0...
ows?service=WMS&version=1.3.0...
ows?service=WMS&version=1.3.0...
http://localhost:3001/collections/wisselstandsensoren/items
Headers
Preview
Response
Initiator
Timing
{bbox: [4.44973900093371, 51.931170004589, 5.10590200161786, 52.1124090047
bbox: [4.44973900093371, 51.931170004589, 5.10590200161786, 52.112409004
0: 4.44973900093371
1: 51.931170004589
2: 5.10590200161786
3: 52.1124090047034
crs: {type: "name", properties: {name: "http://www.opengis.net/def/crs/O
properties: {name: "http://www.opengis.net/def/crs/OGC/1.3/CRS84"}
name: "http://www.opengis.net/def/crs/OGC/1.3/CRS84"
type: "name"
features: [{id: "wisselstandsensoren.2",…}, {id: "wisselstandsensoren.4"
links: [{href: "/collections/wisselstandsensoren/items", rel: "self", ty
0: {href: "/collections/wisselstandsensoren/items", rel: "self", type:
href: "/collections/wisselstandsensoren/items"
rel: "self"
title: "this document"
type: "application/geo+json"
numberMatched: 17
numberReturned: 17
storageCrs: {type: "name", properties: {name: "http://www.opengis.net/de
properties: {name: "http://www.opengis.net/def/crs/EPSG/0/28992"}
name: "http://www.opengis.net/def/crs/EPSG/0/28992"
type: "name"
-03T14:20:05.809Z"
totalFeatures: 17

Output: GeoJSON

- Obsolete standard: https://geojson.org/geojson-spec.html#named-crs
- (Geo)JSON is comfortable for Javascript developers
- Broad support

- GML is suggested as an alternative by OGC: http://docs.opengeospatial.org/is/17-069r3/17-069r3.html#_requirements_classes_for_encodings
    - Native support for CRS
    - Bigger files
    - Lesser support in clients

Best practices

- Geonovum Github: https://github.com/Geonovum/testbed-spatial-APIs

And further

- Finishing the demo
    - Download GeoJSON in multiple CRS-es
- Share hurdles and experiences of developing the API
- Content negotiation vs Headers

Questions