



Zur Entwicklung der OGC-API- Standards und deren Einsatz

Remi Koblenzer, interactive instruments GmbH, 2025

Über interactive instruments



- Softwarelösungen für das Management und die Veröffentlichung von Geodaten
- Schwerpunkt auf Webtechnologien und offenen Standards
- Sitz in Bonn seit 1985, 15 Mitarbeiter
- In OGC aktiv seit ca. 2000, v.a. durch Clemens Portele: (Co-) Editor von OGC API - Features, Styles, Routes, CQL2, JSON-FG

Über mich

Remi Koblenzer

- Seit 2000 bei interactive instruments
- Schwerpunkt auf *XtraServer* • AAA Suite, Unterstützung der Implementierungspartnerschaft AAA-Dienste, XtraServer WFS / WMS
- Seit ca. 2017 Projekte und Produkte auf Basis von OGC-APIs, ursprünglich als „Spatial Data on the Web“ begonnen

Über den Vortrag

Titel:

Zur Entwicklung der **OGC-API-Standards** und deren Einsatz

Wie verwende ich einen Web Feature Services (WFS), um Geodaten zu „erfragen“ und zu verarbeiten?

Tipp: Die Entwicklung des Standards begann Ende 1990er Jahre... also noch z.B. vor „Web 2.0“.

Nutzung von Web Feature Services (WFS) in der Praxis

← → ↻ Nicht sicher aaaxs34/aaa-suite/cgi-bin/alkis/vereinf/wfs

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<ExceptionReport xmlns="http://www.opengis.net/ows/1.1"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="2.0.0"
xsi:schemaLocation="http://www.opengis.net/ows/1.1 http://aaaxs34/aaa-
suite/schema/ogc/ows/1.1.0/owsExceptionReport.xsd">
  <Exception exceptionCode="MissingParameterValue" locator="request">
    <ExceptionText> Request rejected due to errors. InternalExceptionCode:
    iiRequiredParmMissing. Reason: Missing parameter "request" in request "" requested
    of WFSRO service. </ExceptionText>
  </Exception>
</ExceptionReport>
```

Elements Console Sources Network

Filter Invert More filters

All Fetch/XHR Doc CSS JS Font Img Media Manifest WS Wasm Other

500 ms 1,000 ms 1,500 ms 2,000 ms 2,500 ms 3,000 ms

Name Headers Preview Response Initiator

wfs

data:image/svg+xml,...

Request URL: http://aaaxs34/aaa-suite/cgi-bin/alkis/vereinf/wfs

Request Method: GET

Status Code: 400 Bad request

Remote Address: 192.168.230.142:80

Referrer Policy: strict-origin-when-cross-origin

Response Headers Raw

Connection: close

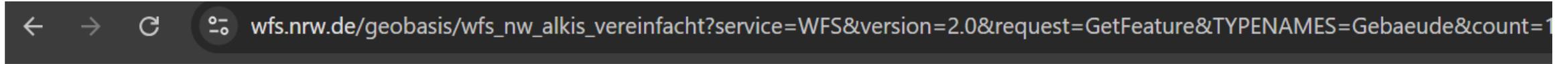
2 requests | 644 B transferred | 7

Console AI assistance

top Filter Default levels No Issues

GET http://aaaxs34/aaa-suite/cgi-bin/alkis/vereinf/wfs 400 (Bad request)

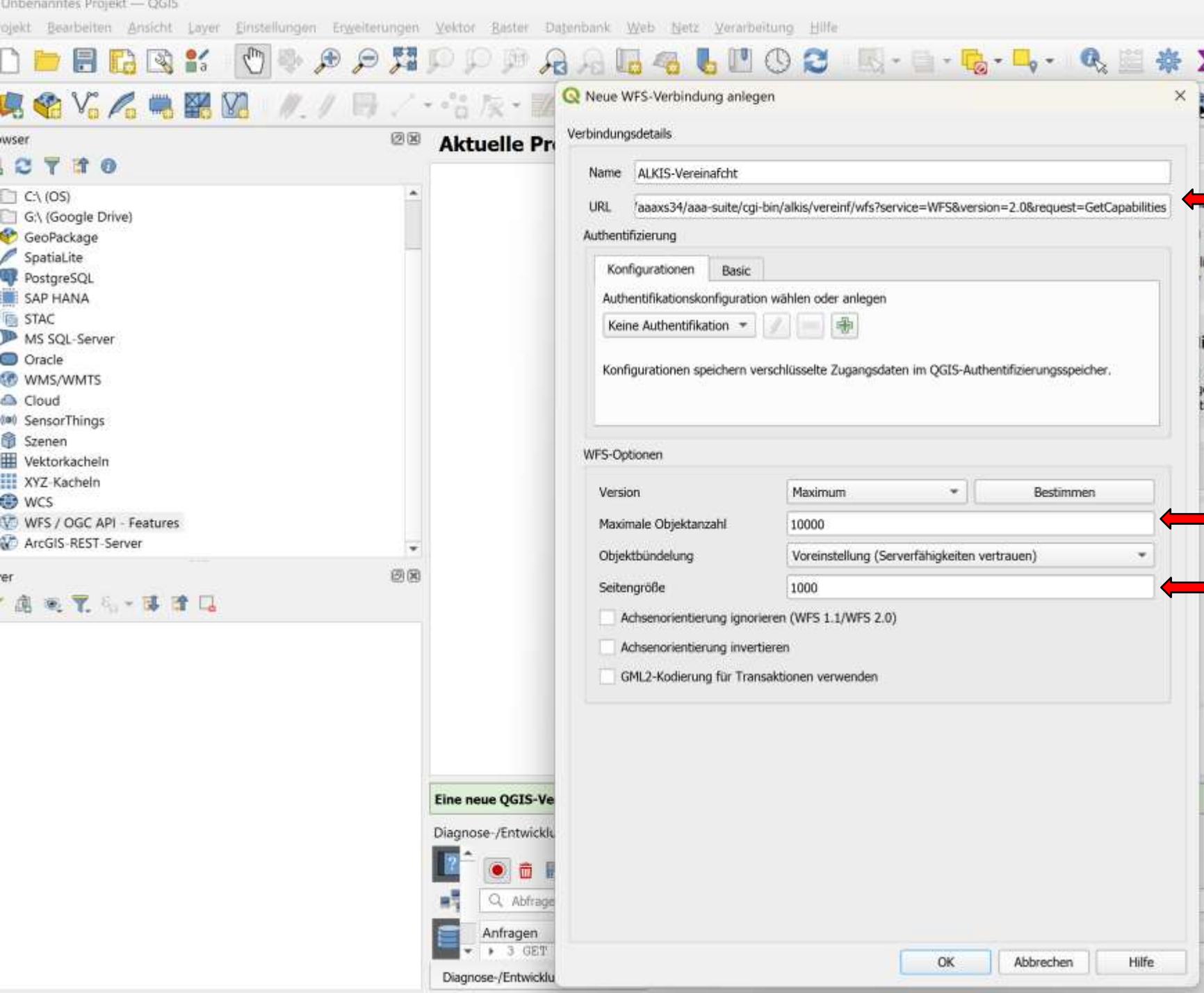
Bei fehlerhaften Parametern passiert auch mal das:



Bad request

Your browser sent a request that this server could not understand.

Der WFS hat hier eigentlich eine Service-Exception im XML-Format geliefert, der Webbrowser zeigt jedoch nur eine generische Fehlerseite an.



URL des WFS

Begrenzung der Objektanzahl, um Überlastung zu vermeiden

-> In diesem Versuch funktionierte die Begrenzung nicht wie erwartet und die Karte blieb leer.

Geht das nicht einfacher und intuitiver?

Daten des Liegenschaftskatasters in NRW

Dieser Dienst stellt Geodaten des Liegenschaftskatasters in einem vereinfachten Datenaustauschschemata über eine Web-API bereit.

Hinweis: Diese API liefert amtliche Koordinaten nur im standardmäßigen Koordinatenreferenzsystem mit dem EPSG-Code 25832 aus. Werden andere unterstützte Koordinatensysteme zur Kartendarstellung gewählt, wird serverseitig eine entsprechende Transformation vorgenommen, die Ungenauigkeiten unterliegt. Für die per Transformation gelieferten nicht amtlichen Koordinaten übernimmt Geobasis NRW keine Gewähr.

Links auf die wichtigsten Ressourcen



[Zu den Daten](#)

[Zu einer Webkarte mit den Daten](#)

[Zu den Daten als Tiles](#)

[Styles zur Darstellung der Daten in Karten](#)

Informationen über die API

API-Beschreibung [Formale Beschreibung der API in OpenAPI 3.0](#)
[Dokumentation der API](#)

API-Anbieter Geobasis NRW
geobasis@bezreg-koeln.nrw.de

Datenlizenz Datenlizenz Deutschland - Zero
<https://www.govdata.de/dl-de/zero-2-0>

Räumlicher Bereich



Daten des Liegenschaftskatasters in NRW

Dieser Dienst stellt Geodaten des Liegenschaftskatasters in einem vereinfachten Datenaustauschschemata über eine Web-API bereit.

Hinweis: Diese API liefert amtliche Koordinaten nur im standardmäßigen Koordinatenreferenzsystem mit dem EPSG-Code 25832 aus. Werden andere unterstützte Koordinatensysteme zur Kartendarstellung gewählt, wird serverseitig eine entsprechende Transformation vorgenommen, die Ungenauigkeiten unterliegt. Für die per Transformation gelieferten nicht amtlichen Koordinaten übernimmt Geobasis NRW keine Gewähr.

Daten

[Flurstück](#) — [weitere Informationen](#) — 9366605 Objekte

Flurstücke sind ein Teil der Erdoberfläche, der von einer im Liegenschaftskataster festgelegten Grenzlinie umschlossen und mit einer Nummer bezeichnet ist. Es ist die Buchungseinheit des Liegenschaftskatasters.

[Flurstückspunkt](#) — [weitere Informationen](#) — 9366605 Objekte

Punktformige Repräsentierung des Flurstücks, wesentliche Angaben zum Flurstück für die Georeferenzierung



[Gebäude, Bauwerk](#) — [weitere Informationen](#) — 12646244 Objekte

Dauerhaft errichtete Gebäude und Bauwerke

[Katasterbezirk](#) — [weitere Informationen](#) — 53557 Objekte

Verwaltungseinheiten des Liegenschaftskatasters (Gemarkungen, Gemarkungsteile/Fluren)

[Nutzung](#) — [weitere Informationen](#) — 5107788 Objekte

Tatsächliche Nutzung

[Nutzung-Flurstück](#) — [weitere Informationen](#) — 14252911 Objekte

Abschnitte der Tatsächlichen Nutzung aus Geometrieverschnidung mit Flurstücken

[Verwaltungseinheit](#) — [weitere Informationen](#) — 455 Objekte

Gemeindeteile, Gemeinden, Kreise / kreisfreie Städte, Regierungsbezirke, Bundesland

Datenlizenz

[Datenlizenz Deutschland - Zero](#)

Metadaten

[Webseite mit weiteren Informationen](#)
[Metadaten zum Datensatz](#)

Weitere Details

Verfügbare

<http://www.opengis.net/def/crs/OGC/1.3/CRS84>

Referenzsysteme

<http://www.opengis.net/def/crs/EPSSG/0/25832>

<http://www.opengis.net/def/crs/EPSSG/0/25833>

Gebäude, Bauwerk

Dauerhaft errichtete Gebäude und Bauwerke

Filter

« 1 »



« 1 »

Gebäude

Objektidentifikator	DENW51AL10007kgBBL
Aktualität	27.01.2009
Bezeichnung	Gebäude
Funktion	Kapelle
Gemeindeschlüssel	05558024

Gebäude, Bauwerk

Dauerhaft errichtete Gebäude und Bauwerke

Filter Apply Cancel

bbox=6.5122,50.8545,7.2192,51.0818 funktion=Kapelle

FIELD

none

filter pattern

Add

Use * as wildcard

Funktion

Kapelle

✓ ✕

BBOX

6,5122

50,9404

Min. Longitude

Min. Latitude

7,2192

51,1673

Max. Longitude

Max. Latitude

✓ ✕

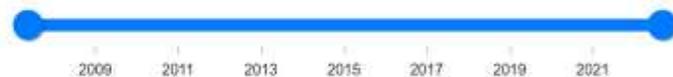
DATE/TIME (UTC)

Period Instant

20.05.2008 12:10:43

07.09.2023 13:27:19

Add



Gebäude

Objektidentifikator	DENW41AL40007rc1BL
Aktualität	04.05.2009
Bezeichnung	Gebäude
Funktion	Kapelle
Gemeindeschlüssel	05362032

Gebäude

Objektidentifikator	DENW41AL1nq0009YBL
Aktualität	22.12.2020
Bezeichnung	Gebäude
Funktion	Kapelle
Gemeindeschlüssel	05362032

Gebäude

Objektidentifikator	DENW41AL40007rDHBL
Aktualität	04.05.2009
Bezeichnung	Gebäude
Funktion	Kapelle
Gemeindeschlüssel	05362032

Gebäude

Objektidentifikator	DENW41AL30005GXWBL
Aktualität	03.05.2009
Bezeichnung	Gebäude
Funktion	Kapelle
Gemeindeschlüssel	05362032

Gebäude

Objektidentifikator	DENW41AL30005GzbBL
Aktualität	03.05.2009
Bezeichnung	Gebäude
Funktion	Kapelle

Eines der Objekte als GeoJSON (f=json):

```

Quelltextformatierung 
{
  "type": "Feature",
  "id": "DENW51AL10007kg8BL",
  "geometry": {
    "type": "MultiPolygon",
    "coordinates": [
      [
        [
          [7.45166217571433, 51.8117378030912],
          [7.45174419216775, 51.8116981942272],
          [7.45175219663106, 51.811696600138],
          [7.45176054668455, 51.811695927785],
          [7.45176895077969, 51.8116962182889],
          [7.45177711983229, 51.8116974408671],
          [7.45178480825944, 51.81169956531],
          [7.45179172820893, 51.8117025248831],
          [7.45179768006319, 51.8117062180459],
          [7.45180243612765, 51.8117105159121],
          [7.4518056239884, 51.811714856073],
          [7.45180759876332, 51.8117194590132],
          [7.45180830676278, 51.8117241981374],
          [7.45180773749044, 51.8117289564113],
          [7.45180589618251, 51.8117335810395],
          [7.451802845153, 51.8117379469542],
          [7.45179866060035, 51.8117419472544],
          [7.45171546279844, 51.8117817384245],
          [7.45166217571433, 51.8117378030912]
        ]
      ]
    ]
  },
  "properties": {
    "aktualit": "2009-01-27T19:06:08Z",
    "gebnuetzbez": "Gebäude",
    "funktion": "Kapelle",
    "fktkurz": "31001_3043",
    "gmdschl": "05558024"
  },
  "links": [
    {
      "href": "https://ogc-api.nrw.de/like/v1/collections/gebaeude_bauwerk/items/DENW51AL10007",
      "rel": "self",
      "type": "application/geo+json",
      "title": "Dieses Dokument"
    }
  ]
}

```

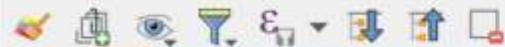


Browser

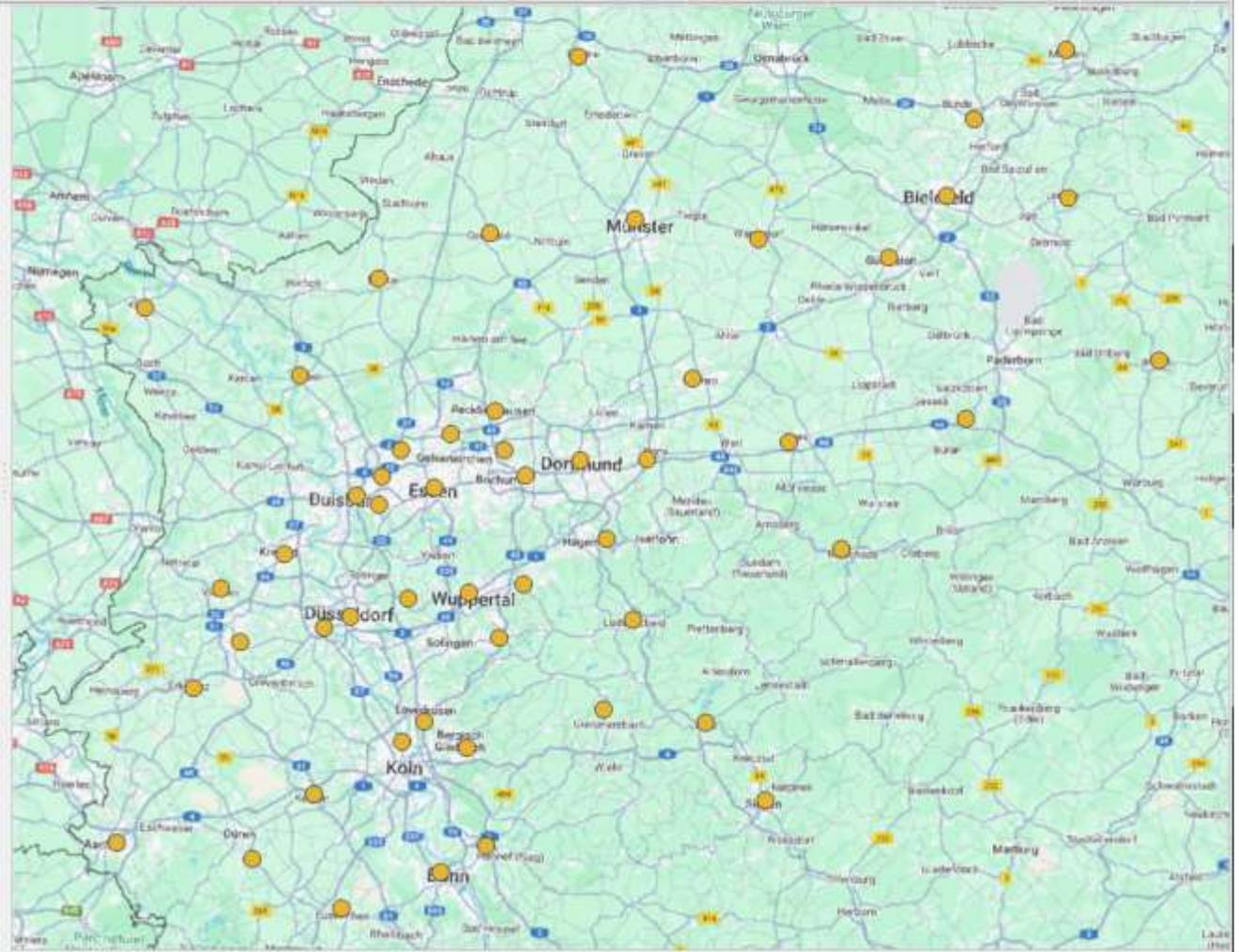


- Oracle
- WMS/WMTS
- Cloud
- SensorThings
- Szenen
- Vektorkacheln
- XYZ-Kacheln
- WCS
- WFS/OGC API Features

Layer

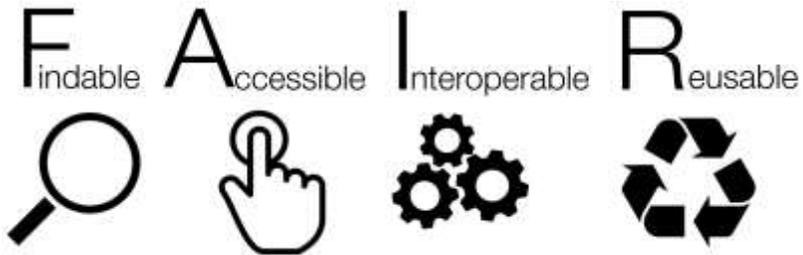


- Feuerwehrleitstellen**
- Google Maps**



Warum und wie sind die OGC-API-Standards entstanden und welche Vorteile bringen sie uns?

FAIR-Prinzipien (2014) → SDW-BP → OGC API - Features



W3C Group Draft Note

TABLE OF CONTENTS

- Abstract
- Status of This Document
- 1. Conformance
- 2. Introduction
- 3. Audience
- 4. Scope
 - 4.1 Spatial data
 - 4.2 Data publication
 - 4.3 FAIR Principles
 - 4.4 Best practice criteria
 - 4.6 Privacy considerations
 - 4.6 Responsible Use
- 5. Best Practices Summary
 - 5.1 Best Practices Summary
- 6. Namespaces
 - 6.1 General remarks
 - 6.2 RDF Namespaces
 - 6.3 XML Namespaces
- 7. Spatial Things, Features and Geometry
- 8. Coverages: describing properties that vary with location (and time)
- 9. Spatial relations
- 10. Coordinate Reference Systems (CRS)
- 11. Linked Data
- 12. Why are traditional Spatial Data Infrastructures not enough?
- 13. The Best Practices
 - 13.1 Web principles for spatial data
 - 13.1.1 Spatial data identifiers
 - 13.1.2 Interable data
 - 13.1.3 Linking data
 - 13.2 Spatial data
 - 13.2.1 Spatial data encoding
 - 13.2.2 Geometries and coordinate reference systems
 - 13.2.3 Relative positioning
 - 13.2.4 Spatial links

Spatial Data on the Web Best Practices

W3C Group Draft Note 19 September 2023



More details about this document

This version:

<https://www.w3.org/TR/2023/DNOTE-20230919-sdw-bp-20230919/>

Latest published version:

<https://www.w3.org/TR/sdw-bp/>

Latest editor's draft:

<https://w3c.github.io/sdw-bp/>

History:

<https://www.w3.org/standards/history/sdw-bp/>
Concise history

Editors:

Jeremy Tandy (Met Office)
Linda van den Brink (Geonovus)
Papam Bernegh (University of Stirling)
Timo Homburg (@hochschule-mainz-university-of-applied-science)

Feedback:

GitHub: [w3c/sdw-bp/issues](https://w3c.github.io/sdw-bp/issues), new issue: [open issue!](https://github.com/w3c/sdw-bp/issues/new)
public-sdw-consortium@w3.org with subject line [sdw-bp] - message: mailto:public-sdw-consortium@w3.org

Contributors:

Phil Archer
Jon Blow
Newbin Cakiran
Byron Cochran
Simon Cox
François Dupont
Andreas Harth
Iza Ham
Bart van Leeuwen
Josh Liebman
Chris Little
Andy Magbit
Peter Parslov
Ed Parsons
Andres Perea
Clemens Portke
Bill Roberts
Lars O. Sæviason
Kerry Taylor
Cok Wele
Rob Allinson
Edward Lewis

OGC Document Number

OGC 15-107



OGC API - Features

Warum sind die „alten“ Dienste-Standards nicht FAIR...: Hürden für Nicht-Experten bei klassischer GDI-Architektur

- Die „klassischen“ Standards WFS, WMS, WCS, WMTS:
 - Sie sind nicht mehr zeitgemäß. Sie basieren auf Technologien von vor über 25 Jahren.
 - Es ist ein komplexes Ökosystem - vor allem nur für Geoexperten.
- Die Erwartungen der Nutzer und der Entwickler haben sich jedoch inzwischen verändert.
- Auch die Verwaltungen haben inzwischen einen anderen Bedarf, unter anderem im Kontext der Bereitstellung ihrer Daten als Open Data.

Wichtige Prinzipien der OGC-API-Standards

1. Gängige Web-Standards und -Technologien als Basis

2. Fokussierung auf "Geo"

The screenshot displays the SwaggerHub interface for an OpenAPI definition. The left sidebar shows navigation options: Info, Tags, Servers, Search, Capabilities, Data, and Schemas. The central code editor shows the following OpenAPI definition:

```
1 openapi: 3.0.2
2 info:
3   title: "A sample API conforming to the draft standard OGC API - Features - Part 1: Core"
4   version: "1.0.0"
5   description: |-
6     This is a sample OpenAPI definition that conforms to the conformance
7     classes "Core", "GeoJSON", "HTML" and "OpenAPI 3.0" of the draft
8     standard "OGC API - Features - Part 1: Core".
9
10    This example differs from the
11    [other example](https://app.swaggerhub.com/apis/cportale/ogcapi-features-1-example1/1.0.0)
12    in that each feature collections is specified explicitly in its own path, not using a path
13    parameter. This API definition is more verbose, but provides information about the feature
14    collection "buildings" (paths "/collections/buildings"), the schema of the building features
15    (schema "buildingGeoJSON") and a filter parameter for building features (parameter "function").
16  contact:
17    name: Acme Corporation
18    email: info@example.org
19    url: "http://example.org/"
20  license:
21    name: CC-BY 4.0 license
22    url: "https://creativecommons.org/licenses/by/4.0/"
23  servers:
24    - url: "https://data.example.org/"
25      description: Production server
26    - url: "https://dev.example.org/"
27      description: Development server
28  tags:
29    - name: Capabilities
30      description: |-
31        essential characteristics of this API
32    - name: Data
33      description: |-
34        access to data (features)
35  paths:
36    '/':
37      get:
38        tags:
39          - Capabilities
40        summary: Landing page
41        description: |-
42          The landing page provides links to the API definition, the conformance
43          statements and to the feature collections in this dataset.
44        operationId: getLandingPage
45        responses:
46          '200':
47            $ref: "https://api.swaggerhub.com/domains/cportale/ogcapi-features-1/1.0.0#/components
```

The right-hand pane displays the rendered API overview:

A sample API conforming to the draft standard OGC API - Features - Part 1: Core

1.0.0 OpenAPI 3.0

This is a sample OpenAPI definition that conforms to the conformance classes "Core", "GeoJSON", "HTML" and "OpenAPI 3.0" of the draft standard "OGC API - Features - Part 1: Core".

This example differs from the [other example](#) in that each feature collections is specified explicitly in its own path, not using a path parameter. This API definition is more verbose, but provides information about the feature collection 'buildings' (paths [/collections/buildings](#)), the schema of the building features (schema [buildingGeoJSON](#)) and a filter parameter for building features (parameter [function](#)).

Acme Corporation - Website
Send email to Acme Corporation
CC-BY 4.0 license

Servers:

Production server

Capabilities: essential characteristics of this API

- GET / landing page
- GET /conformance information about specifications that this API conforms to

Last Saved: 6:54:04 am - Sep 16, 2019 VALID

3. Keine Bindung an ein Format - aktuell: Empfehlung für JSON



4. Unterstützung für HTML und Webbrowser

Home / Daten des Liegenschaftskatasters in NRW / Daten

JSON

Daten des Liegenschaftskatasters in NRW

Dieser Dienst stellt Geodaten des Liegenschaftskatasters in einem vereinfachten Datenaustauschschemata über eine

Hinweis: Diese API liefert amtliche Koordinaten nur im standardmäßigen Koordinatenreferenzsystem mit dem EPSG:31466 aus. Werden andere unterstützte Koordinatensysteme zur Kartendarstellung gewählt, wird serverseitig eine entsprechende Transformation vorgenommen, die Ungenauigkeiten unterliegt. Für die per Transformation gelieferten nicht amtlichen Koordinaten übernimmt Geobasis NRW keine Gewähr.

Daten

- Flurstück** — [weitere Informationen](#) — 9366605 Objekte
Flurstücke sind ein Teil der Erdoberfläche, der von einer im Liegenschaftskataster festgelegten Grenzlinie und mit einer Nummer bezeichnet ist. Es ist die Buchungseinheit des Liegenschaftskatasters.
- Flurstückspunkt** — [weitere Informationen](#) — 9366605 Objekte
Punktförmige Repräsentierung des Flurstücks; wesentliche Angaben zum Flurstück für die Georeferenzierung.
- Gebäude, Bauwerk** — [weitere Informationen](#) — 12646244 Objekte
Dauerhaft errichtete Gebäude und Bauwerke.
- Katasterbezirk** — [weitere Informationen](#) — 53557 Objekte
Verwaltungseinheiten des Liegenschaftskatasters (Gemarkungen, Gemarkungsteile/Fluren).
- Nutzung** — [weitere Informationen](#) — 5107788 Objekte
Tatsächliche Nutzung.
- Nutzung-Flurstück** — [weitere Informationen](#) — 14252911 Objekte
Abschnitte der Tatsächlichen Nutzung aus Geometrieerschneidung mit Flurstücken.
- Verwaltungseinheit** — [weitere Informationen](#) — 455 Objekte
Gemeindeteile, Gemeinden, Kreise / kreisfreie Städte, Regierungsbezirke, Bundesland.

Datenlizenz

[Datenlizenz Deutschland - Zero](#)

Metadaten

[Webseite mit weiteren Informationen](#)
[Metadaten zum Datensatz](#)

Weitere Details

The screenshot shows a REST client interface with a GET request to `/collections/flurstueck/items`. The response is a JSON document containing features in the collection. The response is a document consisting of features in the collection. The features included in the response are determined by the server based on the query parameters of the request.

To support access to larger collections without overloading the client, the API supports paged access with links to the next page, if more features are selected than the page size.

The `bbox` and `datetime` parameter can be used to select only a subset of the features in the collection (the features that are in the bounding box or time interval). The `bbox` parameter matches all features in the collection that are not associated with a location, too. The `datetime` parameter matches all features in the collection that are not associated with a time interval, too. The `limit` parameter may be used to control the subset of the selected features. Each page may include information about the number of features that were not returned (`numberMatched` and `numberReturned`) as well as a link to the next page (`next`).

See the details of this operation for a description of additional query parameters.

Name	Description
<code>access_token</code>	Set the authorization token for the request. If no token is provided, the standard HTTP rules apply, i.e., the authorization header is determined by the request.
<code>bbox</code>	Only features that have a geometry that intersects the bounding box are selected. The bounding box is provided as follows: <code>[minx, miny, maxx, maxy]</code> .

Flur	53
Flurstücksnummer-Zähler	47
Flurstücksnummer-Nenner	1
Regierungsbezirk	Münster
Kreis	Coesfeld
Gemeinde	Lüdinghausen
Flurstück 51/2 in Flur 49, Gemarkung Lüdinghausen-Kirchspiel, Gemeinde Lüdinghausen	
Objektidentifikator	DENW51AL1000718RFL
Aktualität	27.10.2022
ID des Flurstücks	DENW51AL1000718K
Flächeninhalt in m²	39707.00
Flurstückskenzeichen	055100049000510002_
Land	Nordrhein-Westfalen
Gemarkung	Lüdinghausen-Kirchspiel
Flur	49
Flurstücksnummer-Zähler	51
Flurstücksnummer-Nenner	2
Regierungsbezirk	Münster
Kreis	Coesfeld
Gemeinde	Lüdinghausen
Flurstück 28 in Flur 50, Gemarkung Lüdinghausen-Kirchspiel, Gemeinde Lüdinghausen	
Objektidentifikator	DENW51AL1000718BFL
Aktualität	27.10.2022

5. Modulare Bausteine; Fokus auf Fähigkeiten, die viele benötigen

- Start mit Anforderungen, die alle haben
- Weitere Teile, mit spezifischen Anforderungen, die „nur“ viele haben
- → Schnellerer Standardisierungsprozess, auf GitHub
- Beispiel - OGC API Features:
 - Teil 1: Core
 - Lesezugriff, WGS84, einfache Filterung, kein Schema
 - Teil 2: Coordinate Reference Systems by Reference
 - Unterstützung über WGS84 hinaus für gängige Koordinatenreferenzsysteme
 - Teil 3: Filtering + Common Query Language (CQL2)
 - Filterausdrücke für Features aus einer Collection
 - Teil 4: Create, Replace, Update and Delete
 - Aktualisieren von Features (ein Feature pro Request)
 - Teil 5: Schemas
 - Schemainformationen zu den Features einer Collection
 - ... weitere Erweiterungen (6 bis 10)



Beispiel für Verwendung von modularen Bausteinen: ALKIS-Vereinfacht-API implementiert Common, Features, Tiles, Styles

[Home](#) / [Daten des Liegenschaftskatasters in NRW](#)

[JSON](#)

Daten des Liegenschaftskatasters in NRW

Dieser Dienst stellt Geodaten des Liegenschaftskatasters in einem vereinfachten Datenaustauschschemata über eine Web-API bereit.

Hinweis: Diese API liefert amtliche Koordinaten nur im standardmäßigen Koordinatenreferenzsystem mit dem EPSG-Code 25832 aus. Werden andere unterstützte Koordinatensysteme zur Kartendarstellung gewählt, wird serverseitig eine entsprechende Transformation vorgenommen, die Ungenauigkeiten unterliegt. Für die per Transformation gelieferten nicht amtlichen Koordinaten übernimmt Geobasis NRW keine Gewähr.

Links auf die wichtigsten Ressourcen

[Zu den Daten](#)

[Zu einer Webkarte mit den Daten](#)

[Zu den Daten als Vector Tiles](#)

[Styles zur Darstellung der Daten in Karten](#)

Informationen über die API

API-Beschreibung [Formale Beschreibung der API in OpenAPI 3.0](#)
[Dokumentation der API](#)

API-Anbieter Geobasis NRW
geobasis@bezreg-koeln.nrw.de

Datenlizenz Datenlizenz Deutschland - Zero
<https://www.govdata.de/dl-de/zero-2-0>

Räumlicher Bereich



Common Part 2,
Features Parts 1 bis 3

Tiles, Part 1

Styles,
Part 1

Common Part 1

OGC-API-Standards – Überblick (Auswahl)

Discrete Global Grid Systems



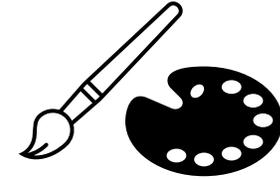
Records



Maps



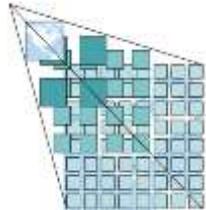
Styles



3D GeoVolumes

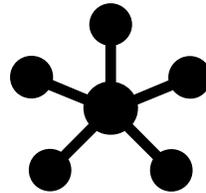


Tiles

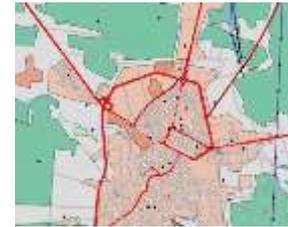


Tile Matrix Set

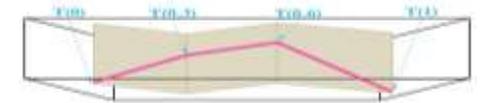
Common



Routes

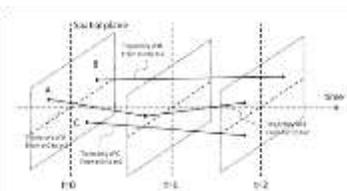


Environmental Data Retrieval

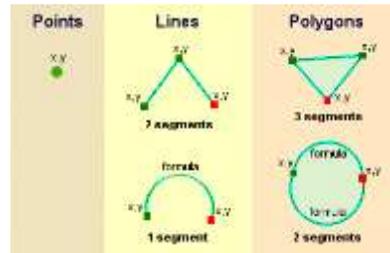


Trajectory

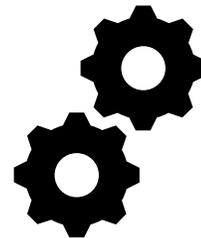
Moving Features



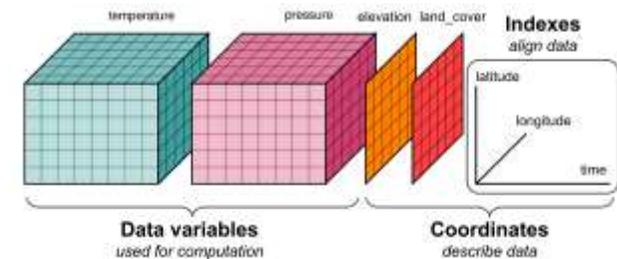
Features



Processes



Coverages



Dicker Rahmen bedeutet mindestens ein Teil ist **verabschiedet**; nicht alle OGC API Standards sind abgebildet (2023)

OGC-API-Standards

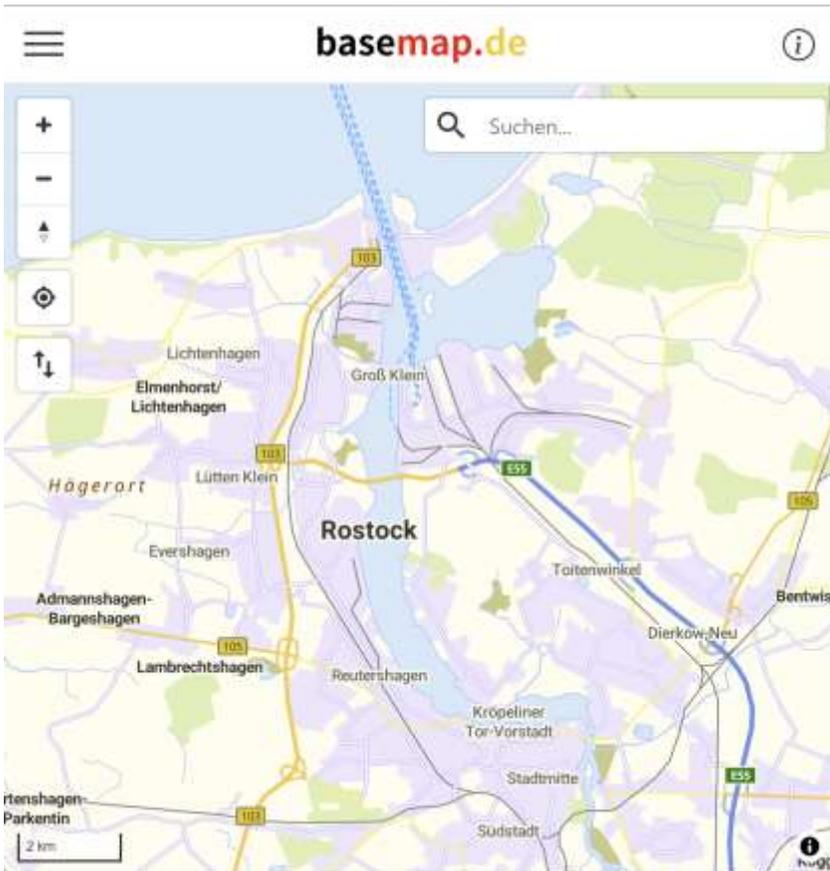
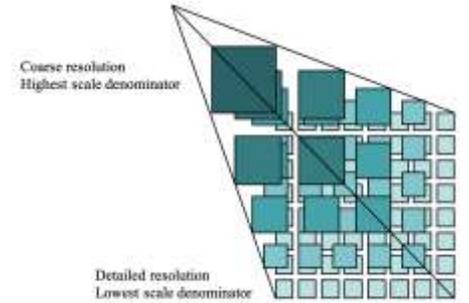
Auswahl: Features, Tiles, JSON-FG

OGC API - Features

```
{
  "type": "FeatureCollection",
  "numberReturned": 10,
  "numberMatched": 52,
  "timeStamp": "2025-05-07T12:07:55Z",
  "features": [
    {
      "type": "Feature",
      "id": "LtS08",
      "geometry": {
        "type": "Point",
        "coordinates": [7.29811817537989, 51.4889309184315]
      },
      "properties": {
        "name": "Bochum",
```

- Zugriff auf Collections,
- Und auf Features („items“) „seitenweise“ auf mehrere oder einzelne Features
- Format: i.d.R. GeoJSON
- Viele Funktionalitäten aus dem WFS, aber oft nur durch die optionalen Erweiterungen (Part 2-10).

OGC API - Tiles



- **Vector Tiles:** Daten, die im Client gerendert werden
- Vielseitiger als Bitmap-Grafiken
- **Kachelschemata** beschreiben, wie der Raum in verschiedenen Maßstabsebenen in ein Gitter unterteilt ist
- Jede Kachel ist identifizierbar über Ebene / Zeile / Spalte
- Ermöglicht Caching, einfacher Zugriff übers Web

OGC API - Maps

- **Maps:** vor allem für Karten gedacht, die vom Server nach einem Style gerendert wurden (wie WMS), auch Luftbilder, Satellitenbilder.
- Format: i.d.R. PNG, JPEG, WebP
- Der Standardisierung von Maps erfährt aktuell nur wenig Aufmerksamkeit.
- Vermutlich zukünftig von untergeordneter Rolle. Gründe:
 - In der Regel für sehr spezielle Nutzung mit hohen kartographischen Ansprüchen sinnvoll.
 - Wie ein WMS erzeugt auch OGC-API-Maps viel Last in der Server-Infrastruktur.
 - Typischerweise längere Bearbeitungszeit und damit clientseitig längere Wartezeit, da das Bild serverseitig gerendert wird.
 - → Dagegen sind Vector Tiles im Web besser nutzbar. Sie enthalten auch schon Daten mit attributiven Informationen, sind viel schneller, dynamisch, und entlasten die Infrastruktur, denn das Rendering erfolgt im Client.

OGC Features and Geometries JSON (kurz: JSON-FG)

- GeoJSON ist weit verbreitet und wird z.B. in OGC API – Features verwendet. Der Standard unterliegt jedoch Einschränkungen, die die Verwendung manchmal stark einschränken. Beispiele:
 - Beschränkung auf WGS-84-Koordinaten
 - Unterstützung von „nur“ der ursprünglichen Geometrietypen „Simple Features“
 - Kein Konzept zur Klassifizierung von Features nach Typ
- **JSON-FG** baut auf GeoJSON-Standard auf und ergänzt ihn mit minimalen Erweiterungen.
 - Unterstützung zusätzlicher Konzepte, die für die breitere Geodaten-Community und die OGC-API-Standards wichtig sind (s. oben – und noch ein paar weitere wie z.B. die Unterstützung von Kreisbögen)
 - JSON-FG ist relevant u.a. für zukünftige Spezifikationen, die die Geobasisdaten verwenden; insbesondere es ist auch ein wichtiges Thema für die Vermessungsverwaltungen.

OGC API - Features - Aktueller Stand und Ausblick

- Part 1: Core v1.0.1
 - → **v1.1 im ersten Halbjahr 2026 erwartet**
[nach JSON-FG, s. vorherige Folie]
- Part 2: Coordinate Reference Systems by Reference v1.0.1
- Part 3: Filtering / Common Query Language (CQL2) v1.0.0
- Part 4 / 5: Create, Replace, Update and Delete / Schemas
 - → **Standards v1.0 noch in 2025 geplant**
- Part 6 / 7 / 8 / 9: Property Selection / Geometry Simplification / Sorting / Text Search → **Standards v1.0 vielleicht ab Mitte 2026**
- Part 10: Search/Queries
 - → **Proposal / Engineering Report; umfangreich, daher dürfte es mit v1.0 noch bis 2027 dauern**

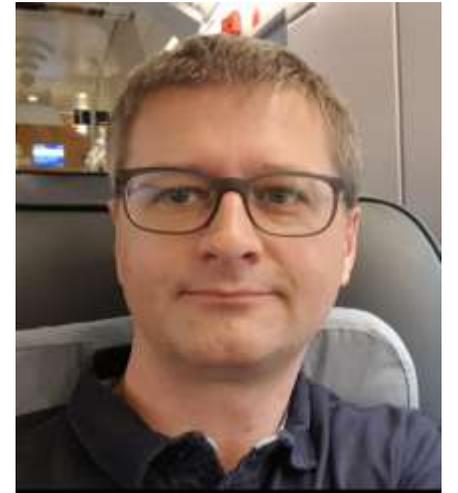


Zusammenfassung

- **Themen im Überblick:**
 - Kurz zur Historie „Von WFS zu modernen Web-APIs“
 - Motivation, Hintergründe und Prinzipien
 - Vorteile der modernen OGC-API-Standards
 - Aktueller Entwicklungsstand von Features und Ausblick
 - Mit Beispielen aus der Praxis



Herzlichen Dank!



Remi Koblenzer
interactive instruments GmbH
koblenzer@interactive-instruments.de