



HAL
open science

Introduction to the RESTful, schemaless, JSON first download service in GeoServer

Oliveira Nuno, Sylvain Grellet, Andrea Aime

► **To cite this version:**

Oliveira Nuno, Sylvain Grellet, Andrea Aime. Introduction to the RESTful, schemaless, JSON first download service in GeoServer. INSPIRE Discovery workshop, Jul 2019, ISPRA, France. <hal-03943569>

HAL Id: hal-03943569

<https://hal.science/hal-03943569v1>

Submitted on 17 Jan 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization

Introduction to the RESTful, schemaless, JSON first download service in GeoServer

Ing. Andrea Aime
Eng. Nuno Oliveira
GeoSolutions



GeoSolutions



- Founded in 2006, offices in Italy & US
- Expertise
 - Image Processing, GeoSpatial Data Fusion
 - Java, Java Enterprise, C++, Python
 - JPEG2000, JPIP, Advanced 2D visualization



- Supporting/Developing FOSS4G projects

- GeoServer, MapStore
- GeoNetwork, GeoNode, Ckan

- Clients

- Public Agencies
- Private Companies

- <http://www.geo-solutions.it>





GeoSolutions
your one-stop-shop for geospatial open source software



INSPIRE
CONFERENCE
2020
DUBROVNIK - CROATIA
Goes virtual 3-12 June

OGC API - Common

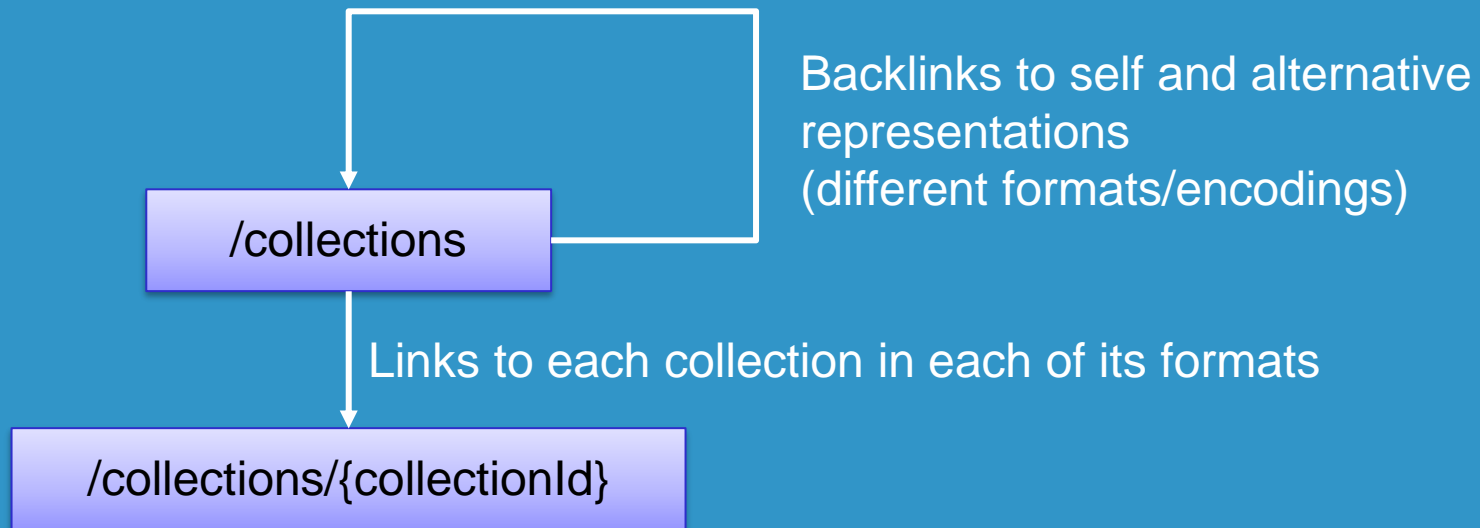


OGC API - Common

- **Can be found at:**
https://github.com/opengeospatial/oapi_common
- **Resources:**
 - / (landing page)
 - /conformance (conformance classes implemented)
 - /api (API definition)
 - /collections (list of collections published)
 - /collection/{collectionId} (metadata of the single collection)
- **Encodings**
 - HTML (for people to browse an API)
 - JSON/GeoJSON

Links, links everywhere!

- All resources are linked to the others, in the various encodings:



```
{  
  "href": "http://data.example.org/collections.json",  
  "rel": "self",  
  "type": "application/json",  
  "title": "this document"  
}
```

Very little is mandatory

- None of the encodings is mandatory, a server could do XML or protocol buffers and still be compliant.
- The `/api` description is not mandatory
- If present, the usage of OpenAPI is suggested but not mandatory!
- A client could work by following links between resources
- How does a client work then?? By checking the conformance declaration at `"/conformance"`:

```
{  
  "conformsTo": [  
    "http://www.opengis.net/spec/ogcapi-features-1/1.0/conf/core",  
    "http://www.opengis.net/spec/ogcapi-features-1/1.0/conf/oas30",  
    "http://www.opengis.net/spec/ogcapi-features-1/1.0/conf/html",  
    "http://www.opengis.net/spec/ogcapi-features-1/1.0/conf/geojson"  
  ]  
}
```



OGC API – Features (Core)



OGC API – Features (Core)

- In addition to OGC API (OAPI) core:
 - `/collection/{collectionId}/items` (features)
 - `/collection/{collectionId}/items/{itemId}`
- Only supported CRS are
 - CRS84 (WGS84 lon/lat)
 - CRS84h (WGS84 lon/lat/height)
- No mandated schema, features can be anything:
 - Simple
 - Complex
 - Heterogeneous
- Extra possible encodings: gmlsf0 (flat) and gmlsf2 (non geometry properties can be nested)

- The OpenAPI document can be setup in whatever way (assuming it matches the spec)
- Two commons approaches:
 - Uniform collections, all collections are described by a /collections/{collectionId} path.
 - Distinct collections, each is given a separate path in the OpenAPI document.

Uniform collections

- **Simpler**
- **Scales to thousands of collections easily**
- **Limited: cannot say anything unique to each collection**
- **GeoServer currently implements this approach**

GET /collections/{collectionId} describe the {collectionId} feature collection

Features Access to data (features).

GET /collections/{collectionId}/items retrieve features of feature collection {collectionId}

GET /collections/{collectionId}/items/{featureId} retrieve a feature; use content negotiation to request HTML or GeoJSON

Distinct collections

- Verbose
- Suitable for small number of collections
- Can provide unique info about a collection, including
 - The returned JSON schema (if so desired)
 - Specific query parameters used for filtering (later)

GET

`/collections` describe the feature collections in the dataset

GET

`/collections/agriculturepnt` describe the Agriculture (Point) feature collection

GET

`/collections/agriculturesrf` describe the Agriculture (Surface) feature collection

GET

`/collections/culturepnt` describe the Culture (Point) feature collection

GET

`/collections/culturesrf` describe the Culture (Surface) feature collection

GET

`/collections/facilitypnt` describe the Facility (Point) feature collection

`/collection/{collectionId}/items`

- Lists the contents of a collection
- Can be a GeoJSON document, GML, anything
- Filtering
 - `bbox` in CRS84 or CRS84h (or specify `bbox-crs`)
 - `datetime` (instant or range of times)
 - extra parameters declared in the API document (by equality)

`.../collections/buildings/items?`

`bbox=41,54,42,55`

`&datetime=2018-02-12T23%3A20%3A52Z`

`&buildingState=good`

Paging

- `limit` query parameter
- A limit is always present (like in WFS 1.x and 2.x) but can be high
- Paging works through links, "prev" and "next" rel values
- Links can be implemented the way you want
- GeoServer uses `offset` as an extra paging parameter

```
[  
  {  
    "href":"http://example.org/collections/buildings/items.json?limit:50",  
    "rel":"prev",  
    "type":"application/geo+json"  
  },  
  {  
    "href":"http://example.org/collections/buildings/items.json?limit:50&offset:50",  
    "rel":"self",  
    "type":"application/geo+json"  
  },  
  {  
    "href":"http://example.org/collections/buildings/items.json?limit:50&offset:100",  
    "rel":"next",  
    "type":"application/geo+json"  
  }  
]
```



GeoSolutions
your one-stop-shop for geospatial open source software



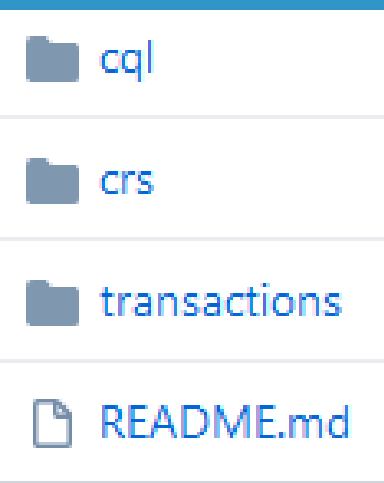
INSPIRE
CONFERENCE
2020
DUBROVNIK - CROATIA
Goes virtual 3-12 June

And that's all
OGC API – Features Core offers!



Extensions!

- Approach similar to GeoPackage.
- **There will be extensions** covering more functionality.
- Some extensions are already **in the making**:
 - **<https://github.com/opengeospatial/ogcapi-features/tree/master/extensions>**
 - CQL filtering
 - CRS (reprojection and filtering)
 - Transactions





GeoSolutions
your one-stop-shop for geospatial open source software



INSPIRE
CONFERENCE
2020
DUBROVNIK - CROATIA
Goes virtual 3-12 June



Géosciences pour une Terre durable
brgm

BRGM boreholes use case



BRGM boreholes use case

- Demo with almost 1 million boreholes:
- Available through:
 - WFS 2.0.0
 - ~~WFS 3.0.0~~
 - **OGC API – Features**
- Simple Features (SF-0) and Complex Features (SF-1)!

Type	Title	Name
	Borehole	eposb:Borehole
	BoreholeView	gsmlp:BoreholeView



Features API landing page

Service Capabilities

FEATURES

1.0

STYLES

1.0

WCS

1.0.0

1.1.0

1.1.1

1.1

2.0.1

WFS

1.0.0

1.1.0

2.0.0

3.0.0

WMS

1.1.1

1.3.0

WPS

1.0.0

TMS

1.0.0

WMS-C

1.1.1

WMTS

1.0.0

← → ↻ 🔒 backoffice-maps.geo-solutions.it/geoserver/ogc/features



GeoServer Features 1.0 Service

This is the landing page of the Features 1.0 service, providing links to the service API and its contents. This document is also available as `application/x-yaml`, `application/json`, `application/xml`, `text/xml`, `application/cbor`.

API definition

The [API document](#) provides a machine processable description of this service API conformant to OpenAPI 3. This API document is also available as `application/x-yaml`, `application/xml`, `text/xml`, `application/cbor`, `text/html`.

Collections

The [collection page](#) provides a list of all the collections available in this service. This collection page is also available as `application/x-yaml`, `application/json`, `application/xml`, `text/xml`, `application/cbor`.

Contact information

- Server managed by -unspecified-
- Organization: -unspecified-
- Mail: -unspecified-



Features API /api

swagger [Explore](#)

Features 1.0 server 2.16-SNAPSHOT OAS3

<https://maps.geo-solutions.it/geoserver/ogc/features/api?f=application%2Fopenapi%2Bjson%3Bversion%3D3.0>

[WFS specification](#)

Server

▾

Capabilities Essential characteristics of this API including information about the data. ▾

- GET** / landing page of this API
- GET** /conformance information about standards that this API conforms to
- GET** /collections describe the feature collections in the dataset
- GET** /collections/{collectionId} describe the {collectionId} feature collection

Features Access to data (features). ▾

- GET** /collections/{collectionId}/items retrieve features of feature collection {collectionId}
- GET** /collections/{collectionId}/items/{featureId} retrieve a feature; use content negotiation to request HTML or GeoJSON

Collections /collections



GeoServer Feature Collections

This document lists all the collections available in the Features service.
This document is also available as [application/x-yaml](#), [application/json](#).

eposb:Borehole

- **Title:** Borehole
- **Geographic extents:**
 - 41.325, -5.15, 51.118, 9.624.
- Data as [HTML](#). Collection items are also available in the following formats:
- Queryables as [HTML](#).

gsmlp:BoreholeView

- **Title:** BoreholeView
- **Geographic extents:**
 - -64.656, -21.388, 139.535, 87.933.
- Data as [HTML](#). Collection items are also available in the following formats:
- Queryables as [HTML](#).

Collection /collections/{collectionId}



GeoServer

eposb:Borehole

- **Title:** Borehole
- **Geographic extents:**
 - 41.325, -5.15, 51.118, 9.624.
- Data as [HTML](#). Collection items are also available in the following formats:
- Queryables as [HTML](#).

Feature schema

- **boreCollarPositionalAccuracy:** Collection
- **metaDataProperty:** Collection
- **relatedSamplingFeature:** Collection
- **length:** Collection
- **__DEFAULT_GEOMETRY__:** Point
- **locatedOnAdminUnit:** Collection
- **georesourceFeature:** Collection
- **shape:** Collection
- **alias:** Collection

--Please choose an option--

--Please choose an option--

- application/ld+json
- application/vnd.google-earth.kml+xml
- application/geo+json
- application/stac+json
- application/gml+xml;version=3.2
- application/json

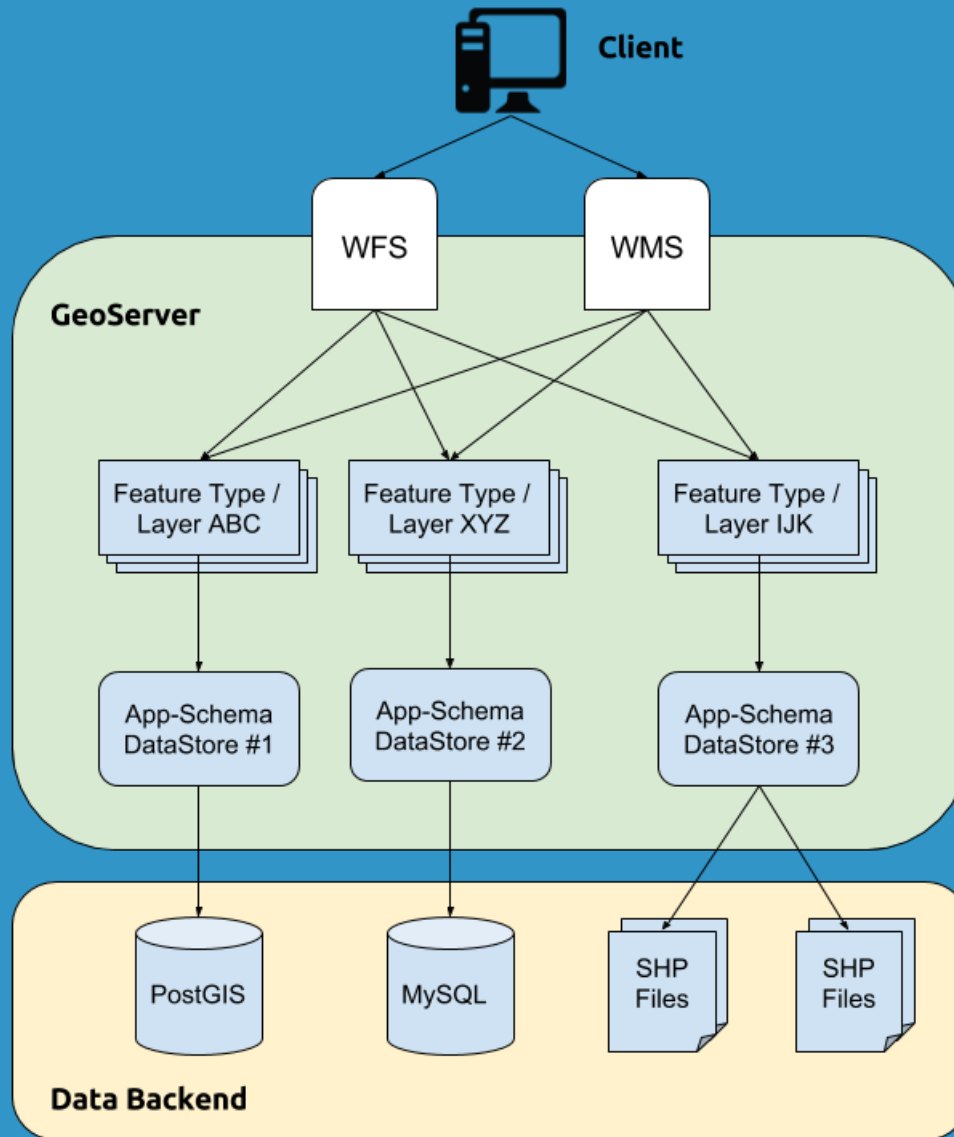
Items /collections/{collectionId}/items

```
{
  type: "FeatureCollection",
  features: [
    {
      type: "Feature",
      id: "0001000001",
      geometry: {
        type: "Point",
        coordinates: [
          [REDACTED],
          [REDACTED]
        ]
      },
      properties: {
        "@featureType": "Borehole",
        identifier: {
          value: "BSS000AAAA",
          "@codeSpace": http://www.ietf.org/rfc/rfc2616
        },
        type: {
          "@href": http://vocabulary.odm2.org/samplingfeaturetype/borehole/,
          "@title": "borehole"
        },
        sampledFeature: [
          {
            "@href": https://sweet.jpl.nasa.gov/2.3/realEarthReference.owl#EarthLithosphere,
            "@title": "Lithosphere"
          }
        ],
        shape: {
          "@dataType": "shape",
          Point: {
            type: "Point",
            coordinates: [
              [REDACTED],
              [REDACTED]
            ]
          }
        },
        custodian: {
          "@dataType": "Custodian",
          custodian: {
            "@dataType": "CI_ResponsibleParty",
            organisationName: "BRGM",
            role: {
              "@codeListValue": "custodian",
              "@codeList": http://www.isotc211.org/2005/resources/Codelist/gmxCodeLists.xml
            }
          }
        }
      }
    }
  ]
}
```

What to do with the existing
mapped and published data sets?

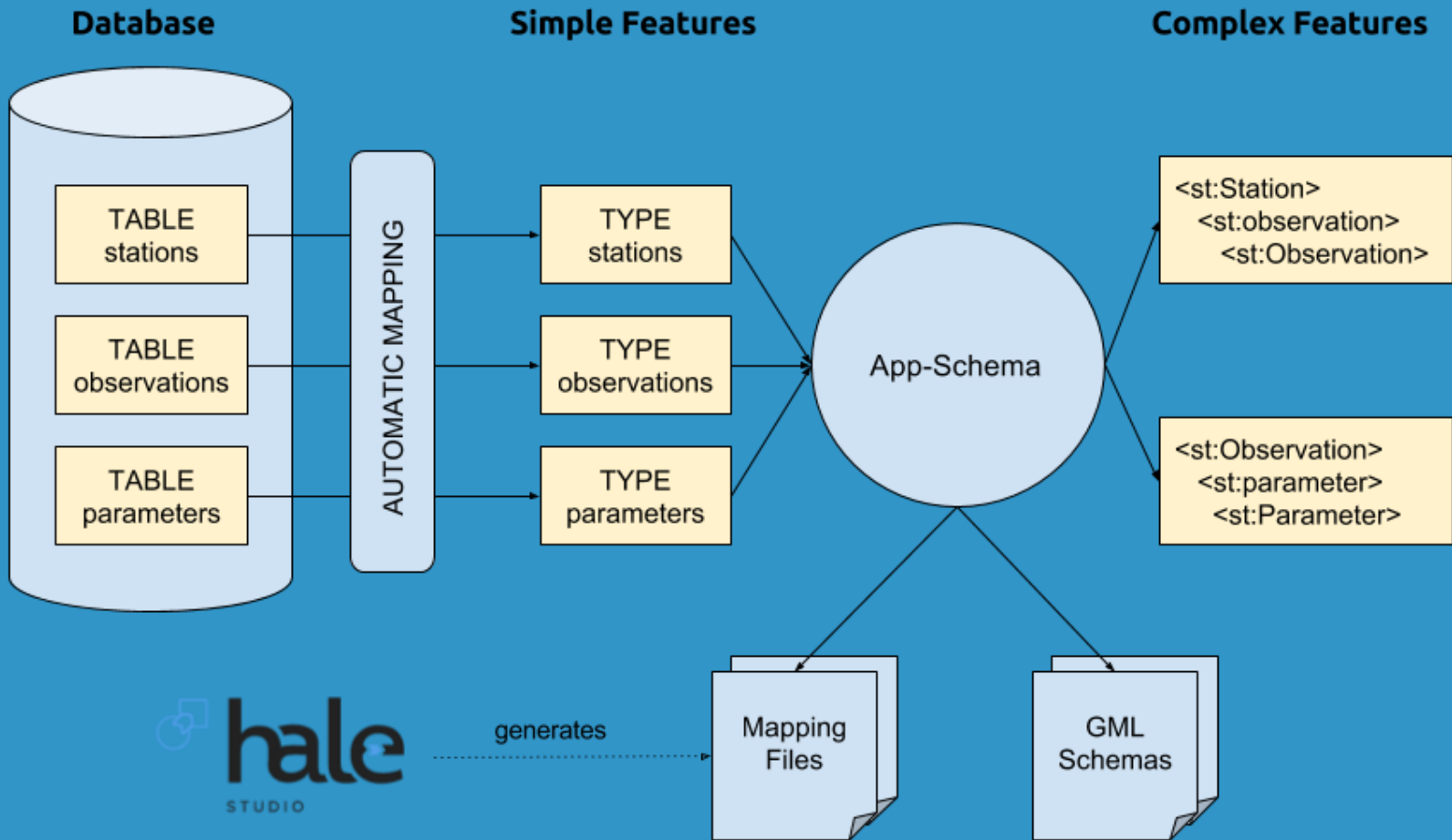


One step back



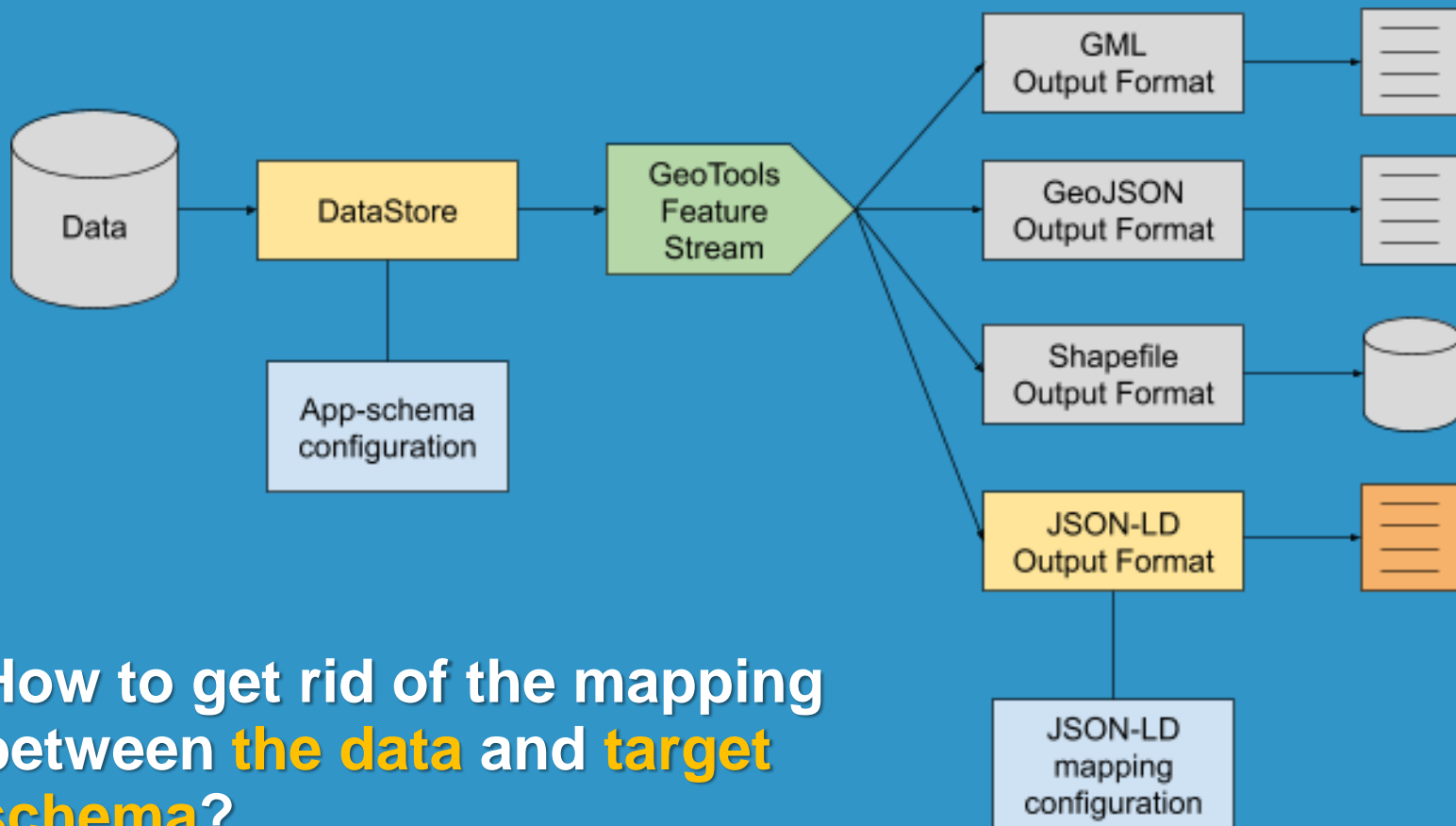


One step back



One step at a time

- One mapping between the data and a target schema
-  **multiple output formats!**



- How to get rid of the mapping between **the data** and **target schema?**

One step at a time

- Why not use templating instead of mappings?



- What You See Is What You Get!

```
"geometry": {
  "@type": "Point",
  "wkt":
  "$${strConcat('<http://www.opengis.net/def/crs/OGC/1.3/CRS84>',toWKT(xpath('eposb:bholeHe
adworks/gwml-wellconstruction:BoreCollar/gwml-wellconstruction:collarLocation'))}"
},
"sam:sampledFeature": {
  "@id": "https://sweet.jpl.nasa.gov/2.3/realmEarthReference.owl#EarthLithosphere",
  "name": "Lithosphere"
},
}
```

```
"eposb:lifeCycleInfo": {
  "$source": "eposb:lifeCycleInfo/eposb:LifeCycleInfo",
  "@type": "LifeCycleInfo",
  "eposb:updateDate": {
    "@type": "time:Instant",
    "time:inXSDDateTime": "${eposb:updateDate/gml:TimeInstant/gml:timePosition}"
  },
  "status": {
    "@id": "${eposb:status/@xlink:href}",
    "name": "Validé"
  },
  "eposb:creationDate": {
    "@type": "time:Instant",
    "time:inXSDDateTime": "${eposb:creationDate/gml:TimeInstant/gml:timePosition}"
  }
},
"eposb:locatedOnAdminUnit": {
  "@id": "${eposb:locatedOnAdminUnit/@xlink:href}",
  "name": "SANGATTE"
},
}
```

One step at a time

- **Filtering using CQL** is supported:
 - `http://.../geoserver/ogc/features/collections/eposb:Borehole/items?f=application/ld+json&limit=50&filter=features.gsmlp:boreholeLength_m.om:amount>83&filter-lang=cql-text`

```

    ▼ geometry: {
      "@type": "Point",
      wkt: "<http://www.opengis.net/def/crs/OGC/1.3/CRS84>POINT (1.35559927237556 51.1020179771059)"
    },
    ▼ "sam:sampledFeature": {
      "@id": https://sweet.jpl.nasa.gov/2.3/realmEarthReference.owl#EarthLithosphere,
      name: "Lithosphere"
    },

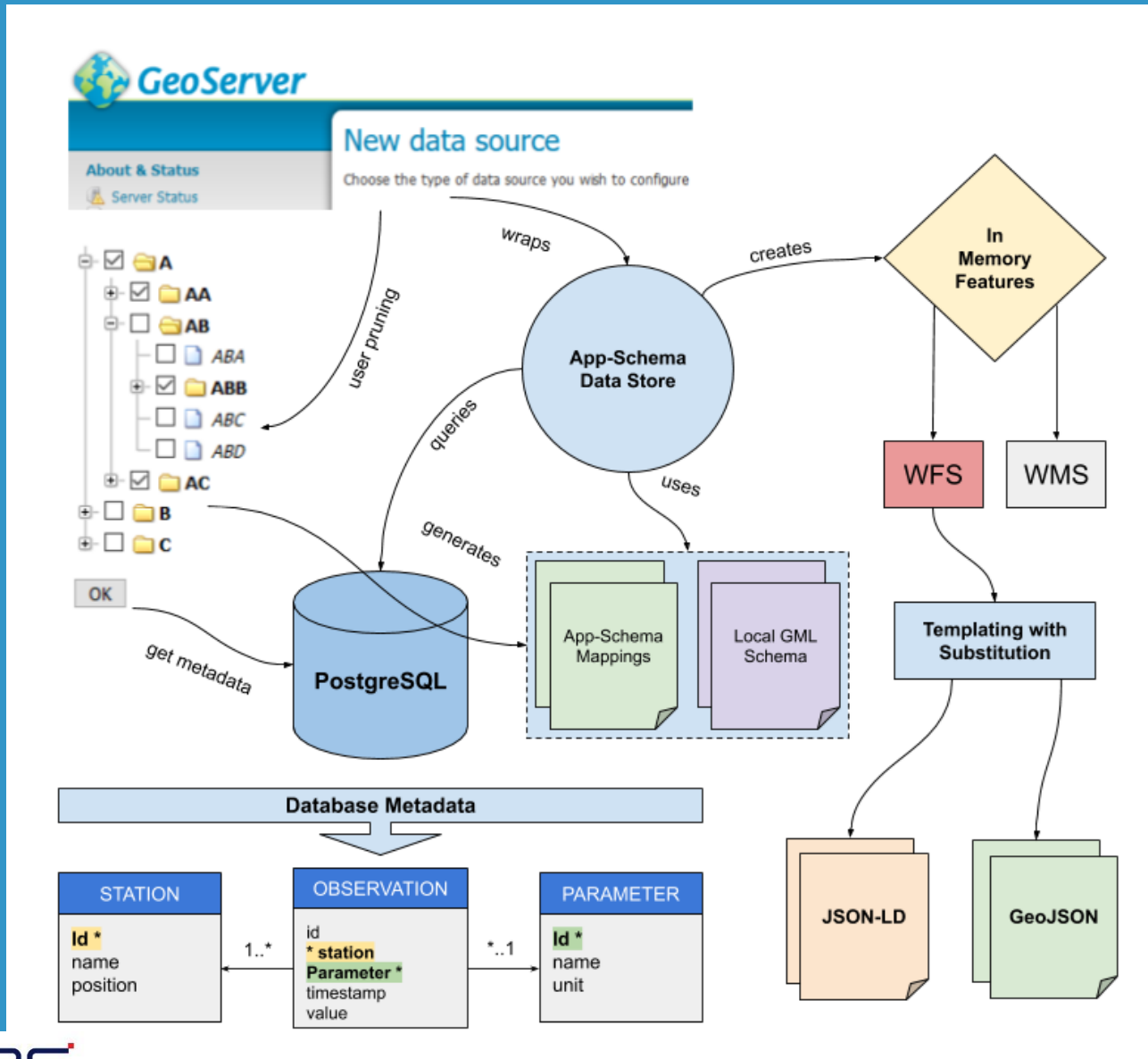
```

```

    ▼ "eposb:lifeCycleInfo": {
      "@type": "LifeCycleInfo",
      ▼ "eposb:updateDate": {
        "@type": "time:Instant",
        "time:inXSDDateTime": "2008/12/357 05:41:34"
      },
      ▼ status: {
        "@id": http://id.eaufrance.fr/nsa/390#XXX,
        name: "Validé"
      },
      ▼ "eposb:creationDate": {
        "@type": "time:Instant",
        "time:inXSDDateTime": "1998/03/70 11:55:17"
      }
    },
    ▼ "eposb:locatedOnAdminUnit": {
      "@id": "6262774",
      name: "SANGATTE"
    },

```

Next steps



That's all folks!



Questions?

info@geo-solutions.it