BBOX – Kompakter OGC API Server für Features, Tiles und mehr

Pirmin Kalberer
@implgeo
Sourcepole AG, Zurich
www.sourcepole.com

SOURCEPOLE
Linux & Open Source Solutions

HAMBURG
FOSSGIS 24

BBOX Services

BBOX services

Composable spatial services.

CI passing | Docker image v2023.03.14.01

```
 ___ ___  _____  __
| _ ) _ )/ _ \ \/ /
| _ \ _ \ (_) >  <
|___/___/\___/_/\_\
```

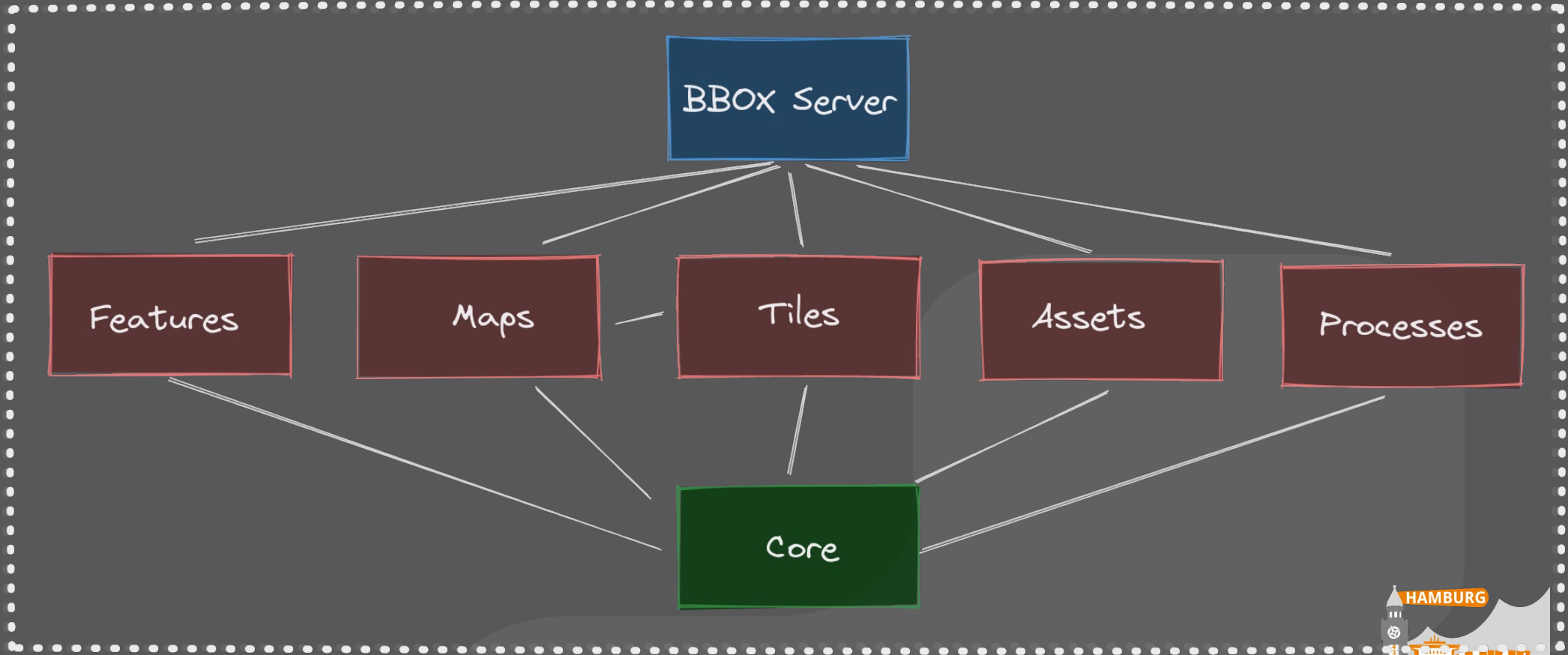
Components:

- BBOX Feature server: OGC API Features service
- BBOX Map server: OGC API Map service
- BBOX Tile server: OGC API Tile service
- BBOX Asset server: Serving static and templated files
- BBOX Processes server: OGC API Processes service

https://bbox.earth/

HAMBURG FOSSGIS 24

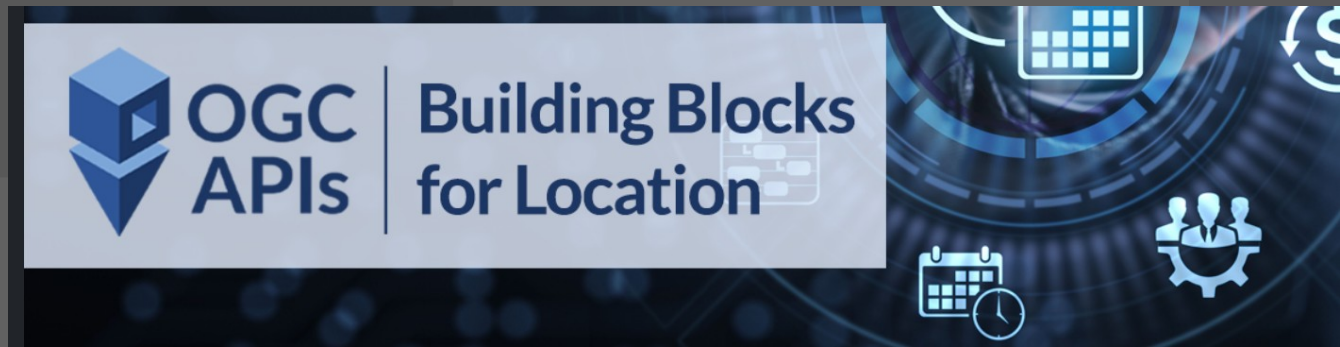
BBOX Services



HAMBURG
FOSSGIS 24

OGC API

- **Standardisierte Endpunkte**
  - Landing page (JSON/HTML) mit Links
  - /conformance
  - /collections
  - /map
  - /tiles
  - /processes
- **OpenAPI Support**



Programmiersprache Rust

- **Wieso Rust?**
  - Performance
  - Sicherheit
  - Produktivität
- **Empowerment**
  - Fearless concurrency – Angstfreie Parallelisierung
  - Langfristige Wartbarkeit von komplexen Projekten
- **Kompiliert auf native Plattformen und WebAssembly**

BBOX Feature Server

- **OGC API - Features - Part 1: Core 1.0**
  - Mit Unterstützung von WFS + WFS-T via QGIS Server
- **JSON + HTML viewer**
- **OpenAPI support**
  - Integriertes Swagger UI
  - Integriertes ReDoc UI
- **Core backends**
  - PostGIS
  - GeoPackage
  - Keine Abängigkeit von externen Bibliotheken

Feature Server - Konfiguration

```
[[datasource]]
name = "mvtbenchdb"
[datasource.postgis]
url = "postgresql://user:pw@127.0.0.1:5439/mvtbench"

[[collection]]
name = "states_provinces_lines"
[collection.postgis]
datasource = "mvtbenchdb"
table_name = "ne_10m_admin_1_states_provinces_lines"
```

- **Auto-Discovery**

```
[[collections.directory]]
dir = "../data"
```

BBOX Map Server

- OGC API – Maps
- Mit Unterstützung von OGC WMS 1.3 Server
- Map Rendering Backends (FCGI)
  - QGIS Server
  - UNN Mapserver
- Backend Dispatcher:
  - Random, Round Robin, WMS Optimized
- Eingebetteter QWC2 Map viewer
- Metrik-Daten für WMS Backends

Map Server - Konfiguration

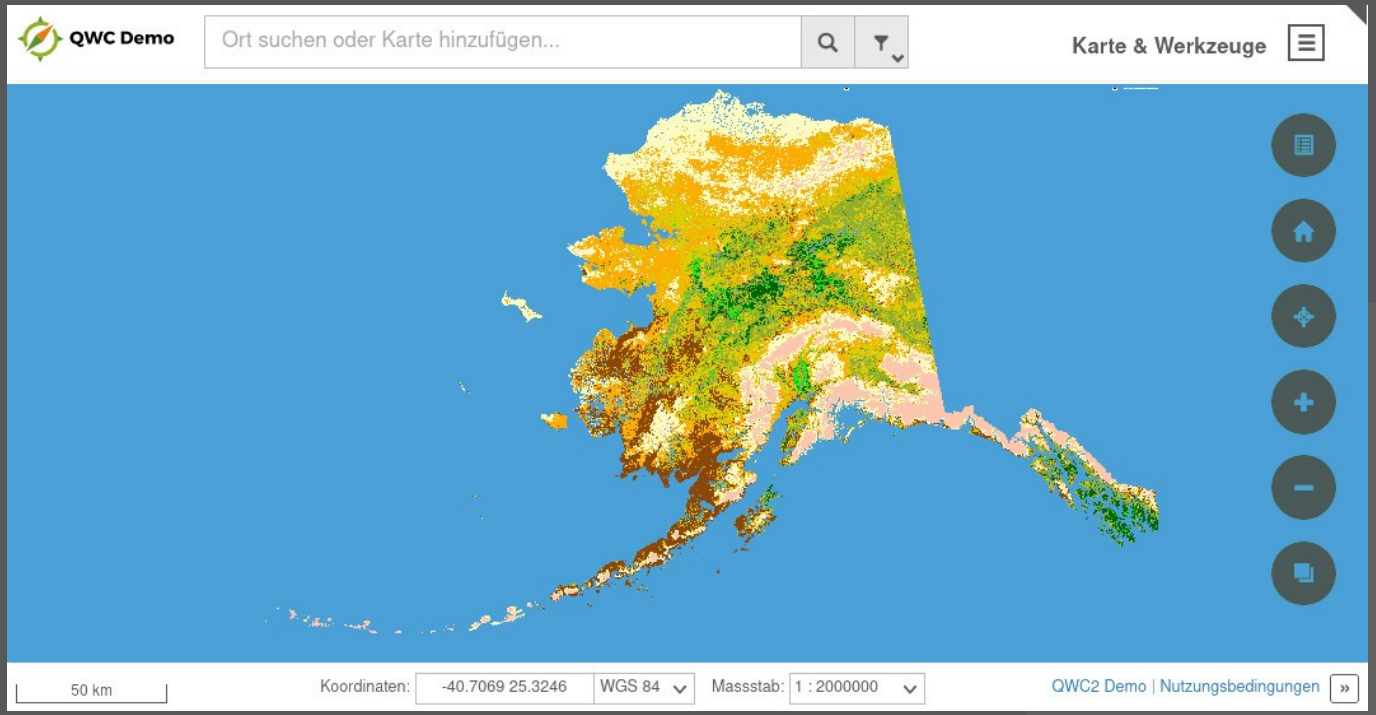
BBOX

```
[mapserver]
num_fcgi_processes = 4

[mapserver.qgis_backend]
project_basedir = "./projects"
qgs.path = "/qgis"                  # base path *.qgs
qgz.path = "/qgz"                   # base path *.qgz

[mapserver.umn_backend]
project_basedir = "./maps"
path = "/wms/map"                   # base path
```

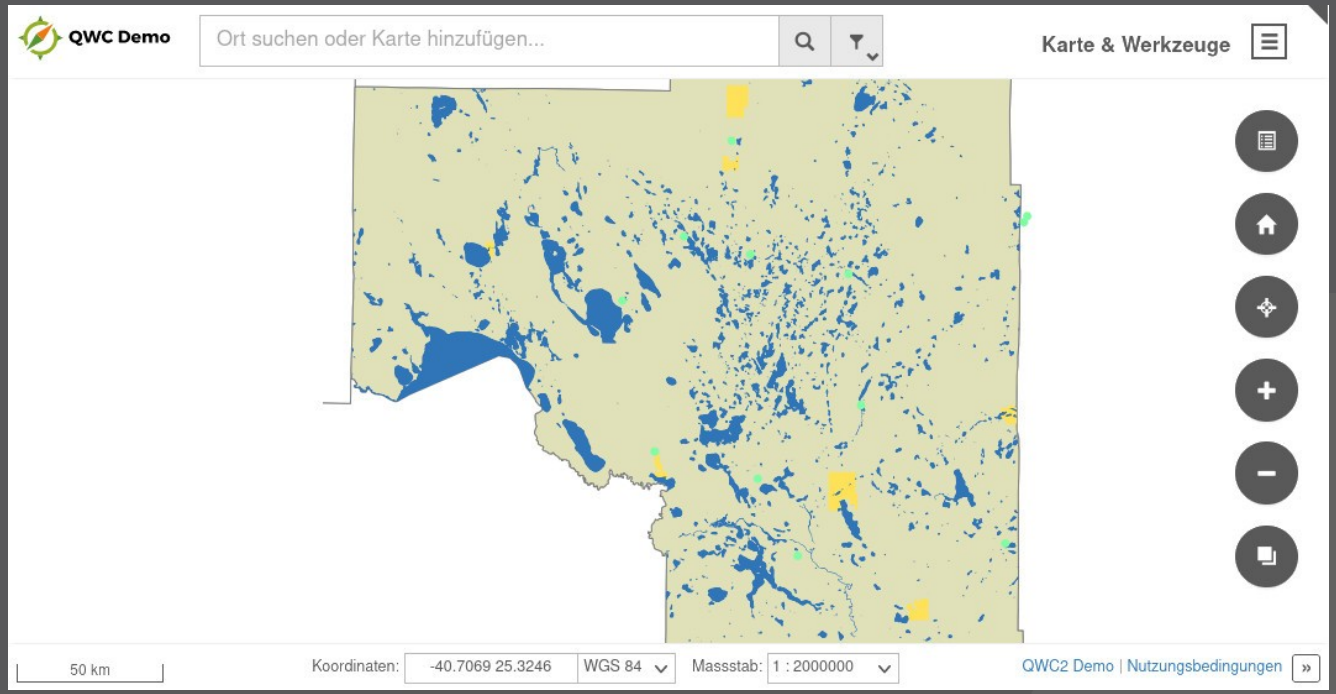
CLI: QGIS Server




```
bbox-server serve –map alaska.qgz
```

HAMBURG
FOSSGIS 24

CLI: UMN Mapserver

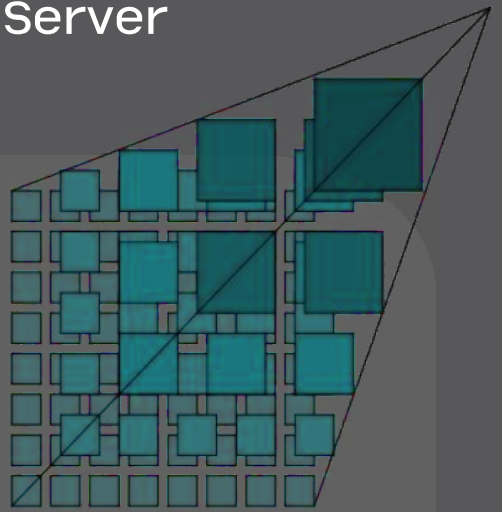



```
bbox-server serve –map itasca.map
```

HAMBURG
FOSSGIS 24

BBOX Tile Server

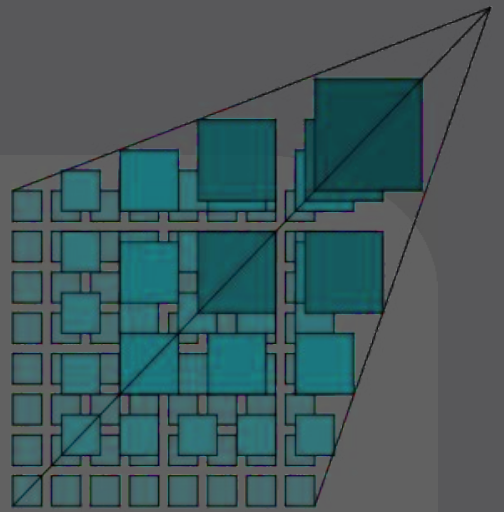
- **OGC API – Tiles - Part 1: Core 1.0**
- **Raster Tile Server**
  - Via Map Service Backends: QGIS Server und MapServer
- **WMS Proxy**
- **Vector Tile Server**
  - Datenquellen: PostGIS, MBTiles, PMTiles
  - XYZ Tileserver Endpunkt (inkl. Tilejson)
- **Parallelisiertes Tile Seeding**
  - Storage: Files, S3, MBTiles, PMTiles



HAMBURG FOSSGIS 24

BBOX Tile Server – Weitere Funktionen

- **Support für benutzerdefinierte Tile Matrix Sets**
  - Beispiel: OSM in Equal Earth Projection
- **Kompatibel mit t-rex Tile Server**
- **Variablen in SQL-Ausdrücken**
  - !bbox!, !zoom!, !x!, !y!, !<uservar>!
- **Diagnostic Tiles**
  - Grösse + Anzahl Features pro Layer



HAMBURG
FOSSGIS 24

Tile Server - Konfiguration

```
[[grid]]
json = "./grids/lv95.json"


[[datasource]]
name = "gebco"
[datasource.wms_proxy]
baseurl = "https://www.gebco.net/service/mapserv?
version=1.3.0"
format = "image/jpeg"


[[tilecache]]
name = "filecache"
[tilecache.files]
base_dir = "/var/tilecache"
```

BBOX

HAMBURG

FOSSGIS 24

Tile Server - Konfiguration

BBOX

```
[[tileset]]
name = "gebco"
cache = "filecache"
wms_proxy = {source = "gebco", layers = "gebco_latest"}

[[tileset]]
name = "ne_countries"
[tileset.postgis]
datasource = "mvtbenchdb"

[[tileset.postgis.layer]]
name = "country-name"
geometry_type = "POINT"
[[tileset.postgis.layer.query]]
sql = "SELECT geom, abbrev, name FROM points"
```

CLI / Kommandozeile

- **Start Tile Server (PostGIS, MBTiles, PMTiles)**

```
bbox-tile-server serve shortbread.mbtiles
```

- **Seed Tile Cache (Files, S3, MBTiles, PMTiles)**

```
bbox-tile-server seed
  –pm-path=shortbread.pmtiles
  –tileset osm
```

BBOX Asset Server

- **Integrierter File Server**
  - Assets: Fonts, Styles, Sprites, …
  - Datenpublikation
- **Templating**
  - Kartenviewer, Story Maps, …
- **QGIS Plugin Repository**

Asset Server - Konfiguration

BBOX

```
[[assets.static]]
# ./assets/* -> http://localhost:8080/assets/
dir = "./assets"
path = "/assets"

[[assets.template]]
dir = "./templates"
path = "/html"

[[assets.repo]]
# QGIS plugin repository
# ./plugins/*.zip ->
http://localhost:8080/qgisrepo/plugins.xml
dir = "./plugins"
path = "/qgisrepo"
```

BBOX Processes Server

- **OGC API – Processes Core**
- **Synchrone und asynchrone Processe**
- **Processing Backend: Dagster**
  - Verschiede Runtime Environments
    - Python
    - Celery, Dask
    - Docker, Kubernetes
  - Support fürPandas, dbt, Spark
  - API: Python, GraphQL
- **Geplantes Backend: Windmill**
  - https://www.windmill.dev/

Authentisierung / Authorisierung

- **Zugriffsgeschützte OGC API services**
- **Geschützte WMS + OGC Maps**
- **Integrierte Provider für Authentisierung**
  - OAuth2 / Openid Connect
  - Geplant: Basic Authentication
- **Nutzung von externen Identity Provider:**
  - Keycloak, Authentik, etc.
  - Multi-Faktor Login, LDAP, SAML2

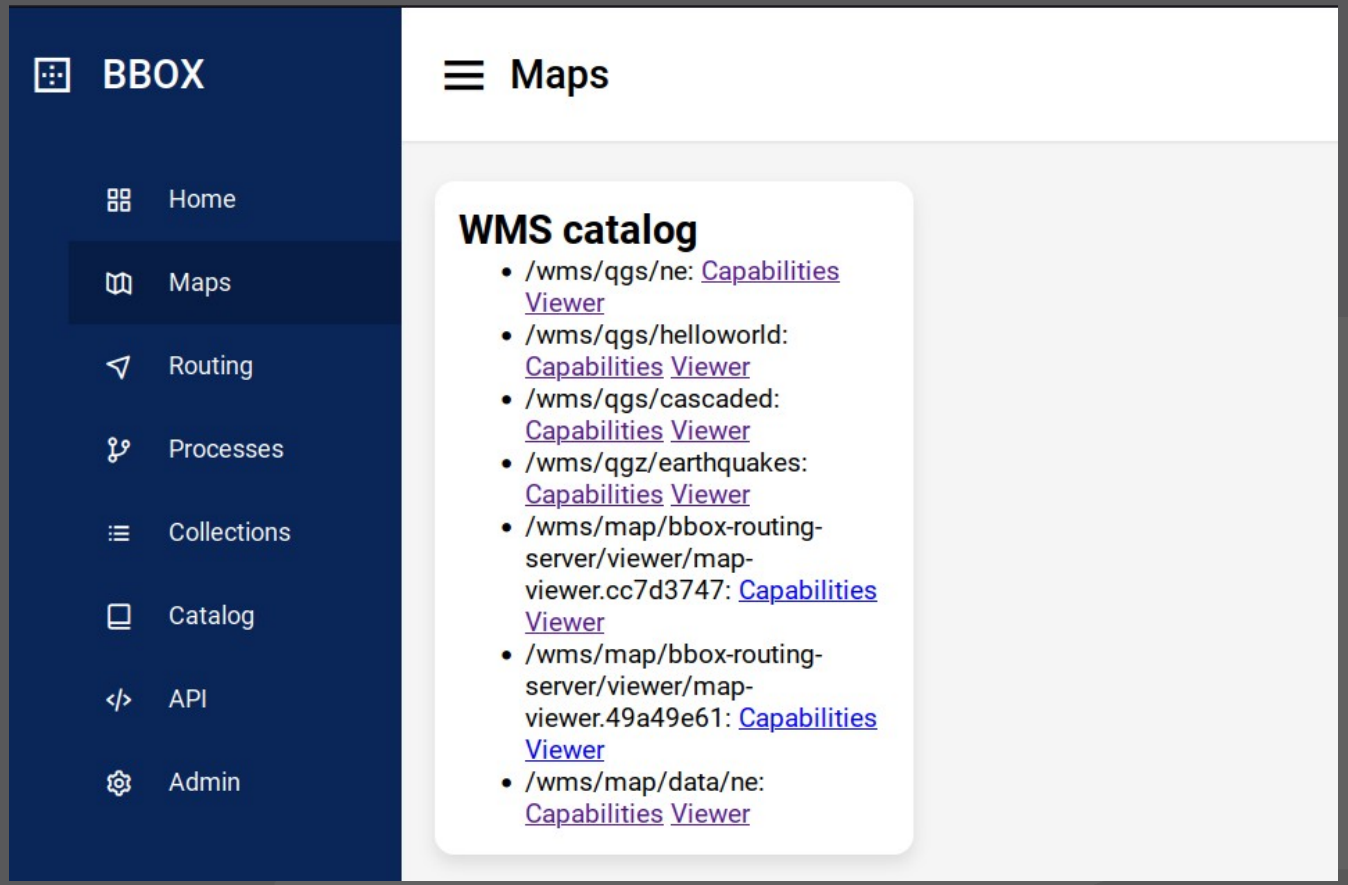
HAMBURG
FOSSGIS 24

Instrumentation + Monitoring

- Prometheus Metriken
- Jaeger Tracing

HAMBURG
FOSSGIS 24

Integriertes Backend (Experimentell)



Modularität

- All-in-one Applikation mit allen Services
- Applikation pro Service
- Docker Container
- Gemeinsame Konfigurationsdatei
- Konfigurierbar über Environment Variablen

Aktueller Status

- **Im produktiven Einsatz:**
  - BBOX Map Server
  - BBOX Processes Server
- **OGC API Conformance Tests:**
  - BBOX Feature Server
  - BBOX Tile Server (OGC Code sprint)
  - BBOX Map Server (OGC Code sprint)
  - (Testsuiten teilweise in Alpha Version oder nicht verfügbar)
- **Experimentell**
  - BBOX Routing Server
- **Version 0.5 Beta 1 zum Download bereit**

Ausblick / Ideen

- **Metadaten Services**
  - STAC, OGC API – Records
- **Volltextsuche**
  - Adressen, Datenfelder, etc.
- **Arrow-Format**
  - Feature-Service im Apache Arrow Format
- **Story Maps**
  - Markdown mit Kartenfunktionen
- **IoT**
  - OGC SensorThings API
- **Mehr**
  - Dashboards, 3D Tiles, Point cloud service (COPC), …

Nächste Schritte

- **Community Building**
  - Feedback von Early Adopters
  - Weitere Kontributoren
  - Verbesserung Dokumentation / Homepage
- **Prioritäten**
  - Release 0.5.0
  - Weitere Funktionen?
  - Weitere Formate?
  - Optimierung für bestimmte Anwendungsszenarien?
  - Beeinflusst durch User, Kunden und Kontributoren

Zusammenfassung

- **Modulare OGC API Services**
- **Enterprise Funktionalität**
  - Instrumentation + Monitoring
  - 1st class Docker support
  - Authentication / Authorization
- **Einfache Nutzung**
  - bbox serve –map alaska.qgz
- **Nutzung von bewährten Raster-Backends**
  - MapServer and QGIS Server
- **Open Source (MIT/Apache)**
  - https://github.com/bbox-services/bbox

Danke!

Pirmin Kalberer
@implgeo@mapstodon.space