

Forschungsprojekt

Visualisierung von erweiterten Geodaten in der PrioBike-App

Markus Wieland // Dresden, 20.09.2022

verantwortlicher Hochschullehrer: Dr.-Ing. Thomas Springer
Betreuer: Dipl.-Inf. Philipp Matthes

Gliederung

- **Daten**
 - OpenStreetMap
 - Geo-Webservices
 - WMS / WFS / WCS
 - Multi-Criteria Decision Analysis
- **Einbindung in die PrioBike App**
 - GeoJSON
 - WMS
 - Kommunikation mit dem Server

Begriffe

Feature ... Geoobjekt

- Abstrahiertes Objekt der realen Welt
- Immer ein *Geometrie* Attribut (Punkt, Polylinie, Polygon, ...)
- Zusätzlich nicht-räumliche Attribute (Fachdaten wie Farbe, Name, Anzahl, ...)

Feature Class ... Sammlung von Features

- Selbe räumliche Repräsentation
- Gemeinsame Attribute
- In GeoJSON „*FeatureCollection*“

OpenStreetMap



- Von Community erstellt
- Frei zugänglich für alle
- Keine Lizenz
- Große Anzahl umfangreicher Datensätze

- Fehlerhafte Daten
- Keine Garantie für Vollständigkeit

Datengrundlage

Attribute

| | |
|-----------------|--|
| <i>OBJECTID</i> | Index innerhalb der FeatureClass |
| <i>osm_id</i> | ID aus dem OpenStreetMap Datensatz |
| <i>shape</i> | Geometrie Attribut |
| <i>fclass</i> | Art des Features (parking_bicycle, ...) |
| <i>name</i> | Name des Ortes sofern vorhanden (Name eines Fahrradladens) |

Datengrundlage

Quelle <https://download.geofabrik.de/europe/germany/hamburg.html>

- gis_osm_pois_free.shp / gis_osm_pois_a_free.shp
- gis_osm_traffic_free.shp / gis_osm_traffic_a_free.shp

Koordinatensystem WGS84

Verarbeitung export_osm_data.py

- Liest Daten aus .shp-Datei
- Filtert nach *fclass* Wert
- Exportiert gefilterte Features als .geojson-Datei

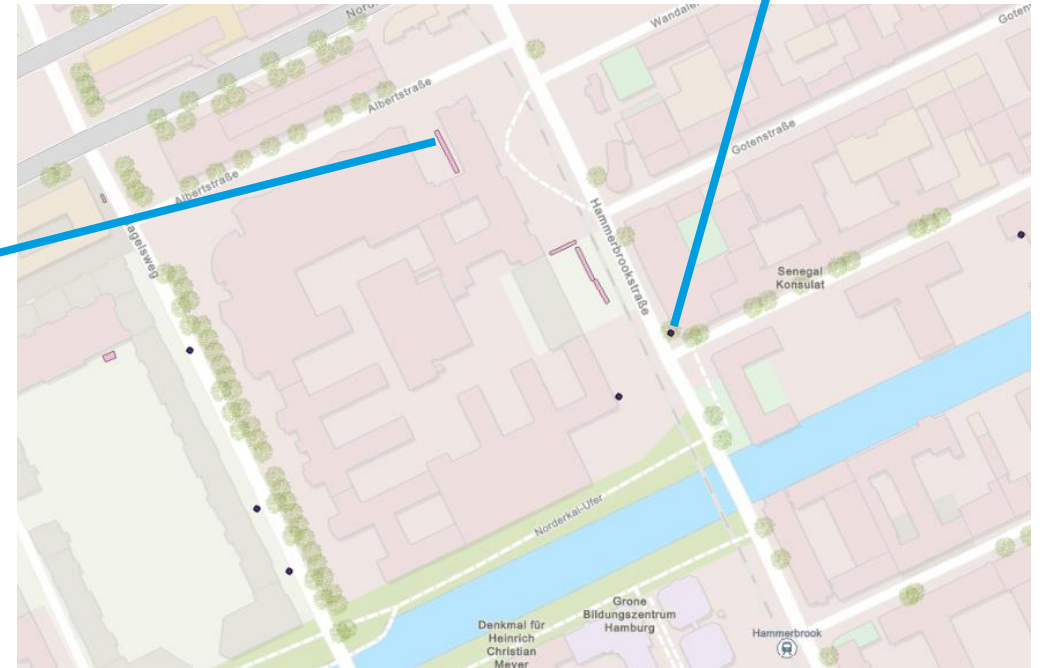
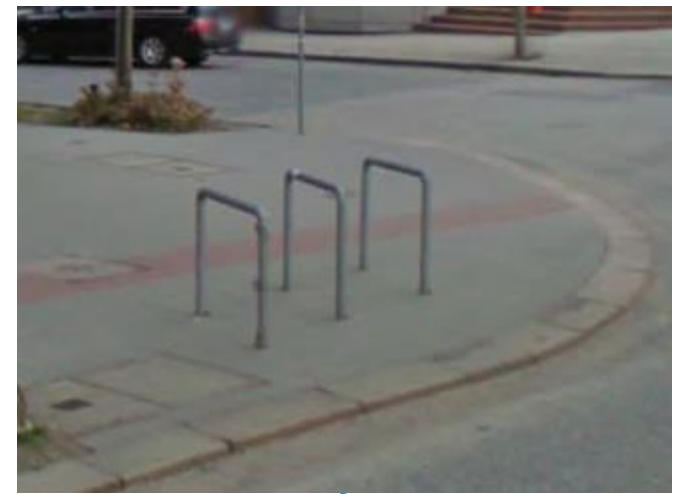
| | code | fclass | name |
|--|------|---------------------|------------------------|
| | 2005 | post_office | |
| | 2205 | playground | |
| | 2907 | camera_surveillance | Shell |
| | 2006 | telephone | |
| | 2501 | supermarket | Marktkauf Wilhelmsburg |
| | 2542 | bicycle_shop | Radsport Wulff |
| | 2601 | bank | Hamburger Sparkasse |
| | 2004 | post_box | |
| | 2907 | camera_surveillance | |
| | 2004 | post_box | |
| | 2601 | bank | Hamburger Sparkasse |

Fahrradständer

/data/generated/osm/bicycle_parking.geojson

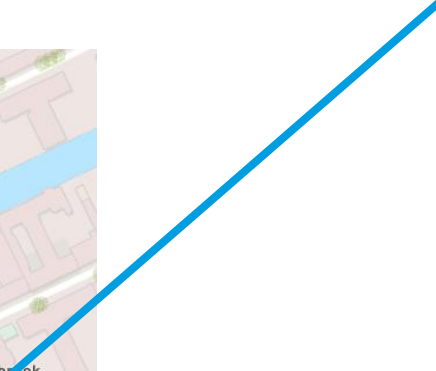
~ 2.600 Punktdaten

~ 1.200 Polygondaten




```
/data/generated/osm/bicycle_rental.geojson
```

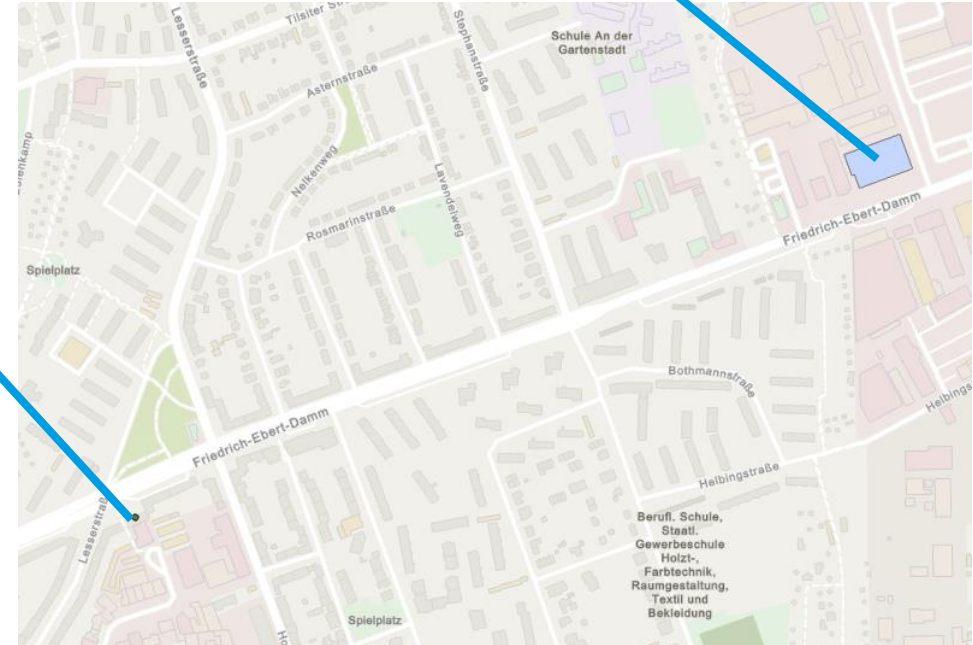
-
- A wide-angle shot of a busy city street. In the foreground, a red car is partially visible. A black Smart car is parked on the left. A blue van is parked further back. Several red delivery vans are parked along the right side of the street. Pedestrians are walking on the sidewalk, and a few bicycles are parked near a metal railing. The background shows a modern building with large windows and a red sign.



Fahrradläden

/data/generated/osm/bicycle_shop.geojson

- 11 Polygondaten
- 145 Punktdaten
- Mögliche Reparatur? -> Ansicht wenn benötigt



Geo-Webservices

Geoportal Hamburg, Deutscher Wetterdienst,
OpenData Portal Hamburg

Geo-Webservices

Standardisierte Bereitstellung von Geodaten

WMS ... **W**eb **M**ap **S**ervice

- Fertige Karten (Bilder)
- GetMap-Request

WFS ... **W**eb **F**eature **S**ervice

- Features
- GetFeature-Request

WCS ... **W**eb **C**overage **S**ervice

- Rasterdaten
- GetCoverage-Request

GetCapabilities

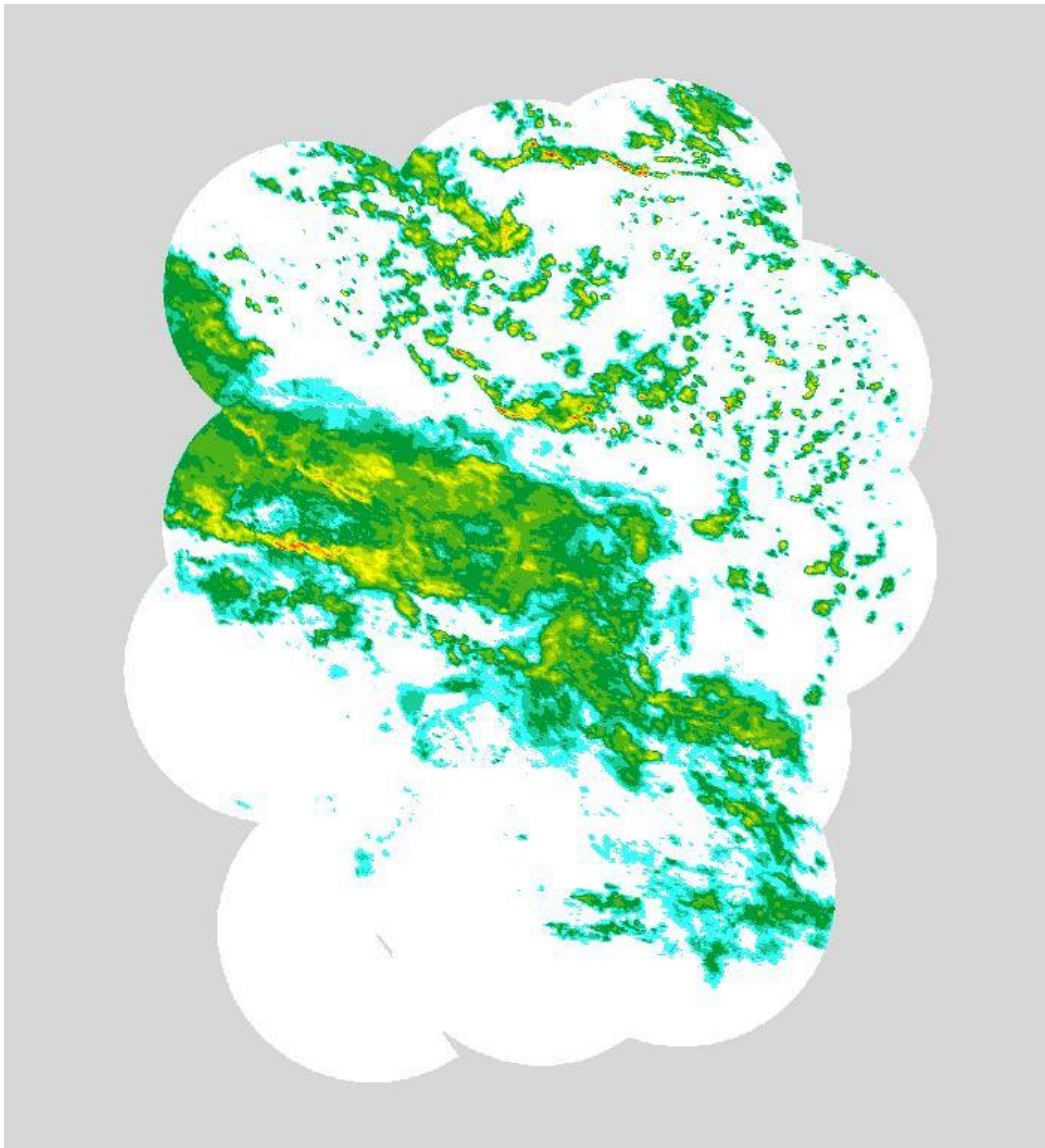
- XML
- Übersicht über Fähigkeiten des Services
 - Verfügbare Datenformate
 - Boundingbox
 - Features/Raster/Map
 - Mögliche Stylings

Anmerkung: Nach WFS Spezifikation, werden Felder deren Wert „null“ ist gar nicht zurückgegeben. Um eine Übersicht über alle Möglichen Attribute zu bekommen, führt man eine „DescribeFeatureType“ Request aus.

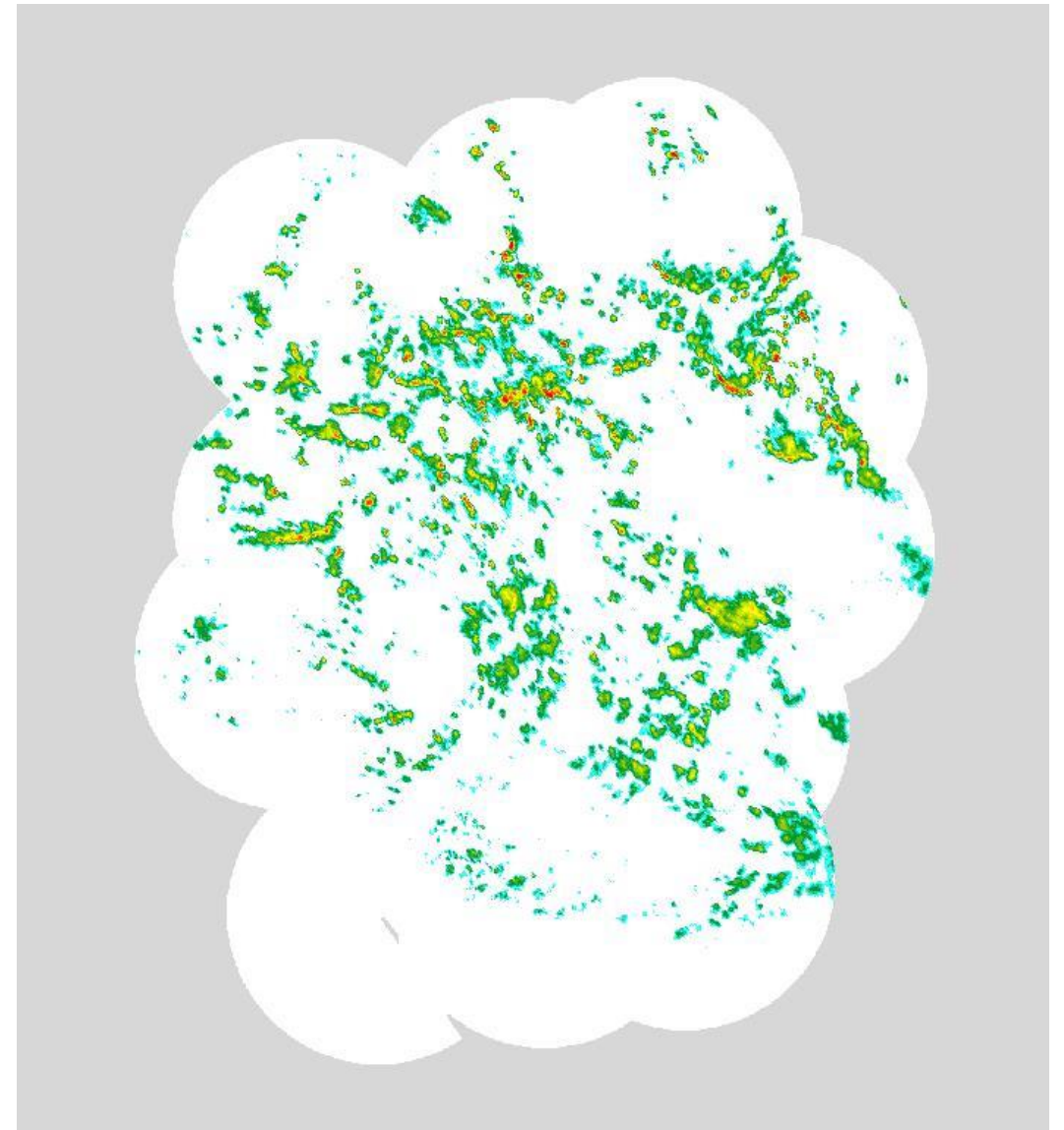
Deutscher Wetter Dienst

<https://maps.dwd.de/geoserver/wms?SERVICE=WMS&VERSION=2.0.0&REQUEST=GetCapabilities>

<https://maps.dwd.de/geoserver/wms?service=WMS&version=1.1.0&request=GetMap&layers=dwd%3ANiederschlagsradar&bbox=-543.462%2C-4808.645%2C556.538%2C-3608.645&width=703&height=768&srs=EPSG%3A1000001&styles=&format=image%2Fpng&time=2022-09-18T12:00:00.000Z>



18.09.2022 12:00



19.09.2022 12:00

StadtRAD Stationen

/data/generated/wfs/stadt_rad.geojson

Beachte Koordinatensystem! Hier wird EPSG:4326 verwendet!

https://geodienste.hamburg.de/HH_WFS_Stadtrad?SERVICE=WFS&REQUEST=GetCapabilities

https://geodienste.hamburg.de/HH_WFS_Stadtrad?SERVICE=WFS&VERSION=2.0.0&REQUEST=GetFeature&typeName=de.hh.up:stadtrad_stationen&outputFormat=application/geo%2bjson&srsname=EPSG:4326

StadtRAD Stationen

/data/generated/wfs/stadt_rad.geojson

Geometrie Punkt

Quelle HH_WFS_Stadtrad

Anzahl 289

Attribute

anzahl_raeder

Anzahl verfügbarer Räder

anzahl_pedelec

Anzahl verfügbarer Pedelecs

uid

ID der Station

stand

Letztes Update der Daten dieser Station

art_der_station

Art der Station

anzahl_bike

Siehe anzahl_raeder ... nur als String?

anzahl_cargobike_electric

Anzahl verfügbarer elektrischer Lastenräder

name

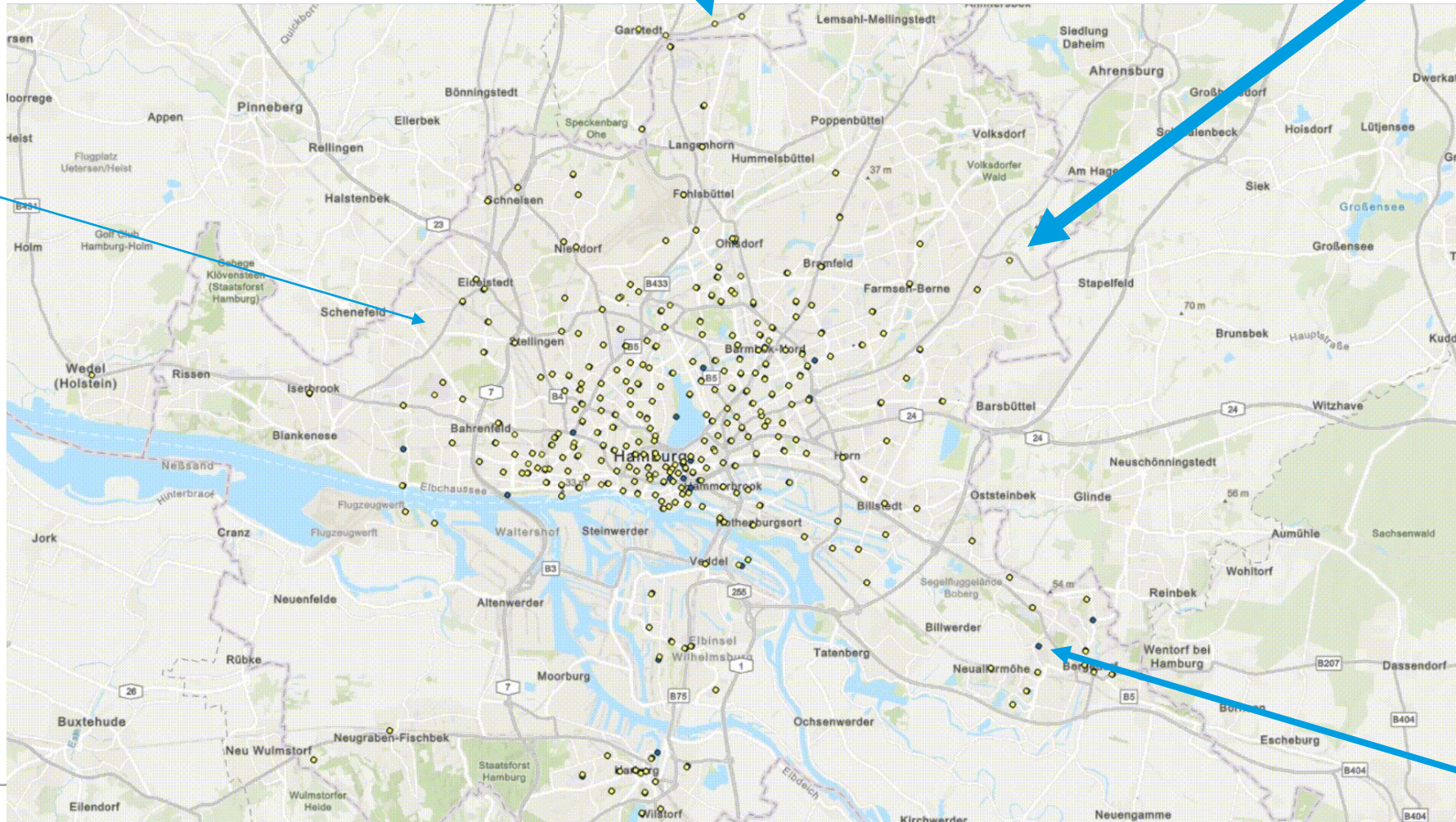
Name der Station



StadtRAD Station / Fahrradleihstationen (OSM)

Vergleich

Gelb ... OpenStreetMap Daten



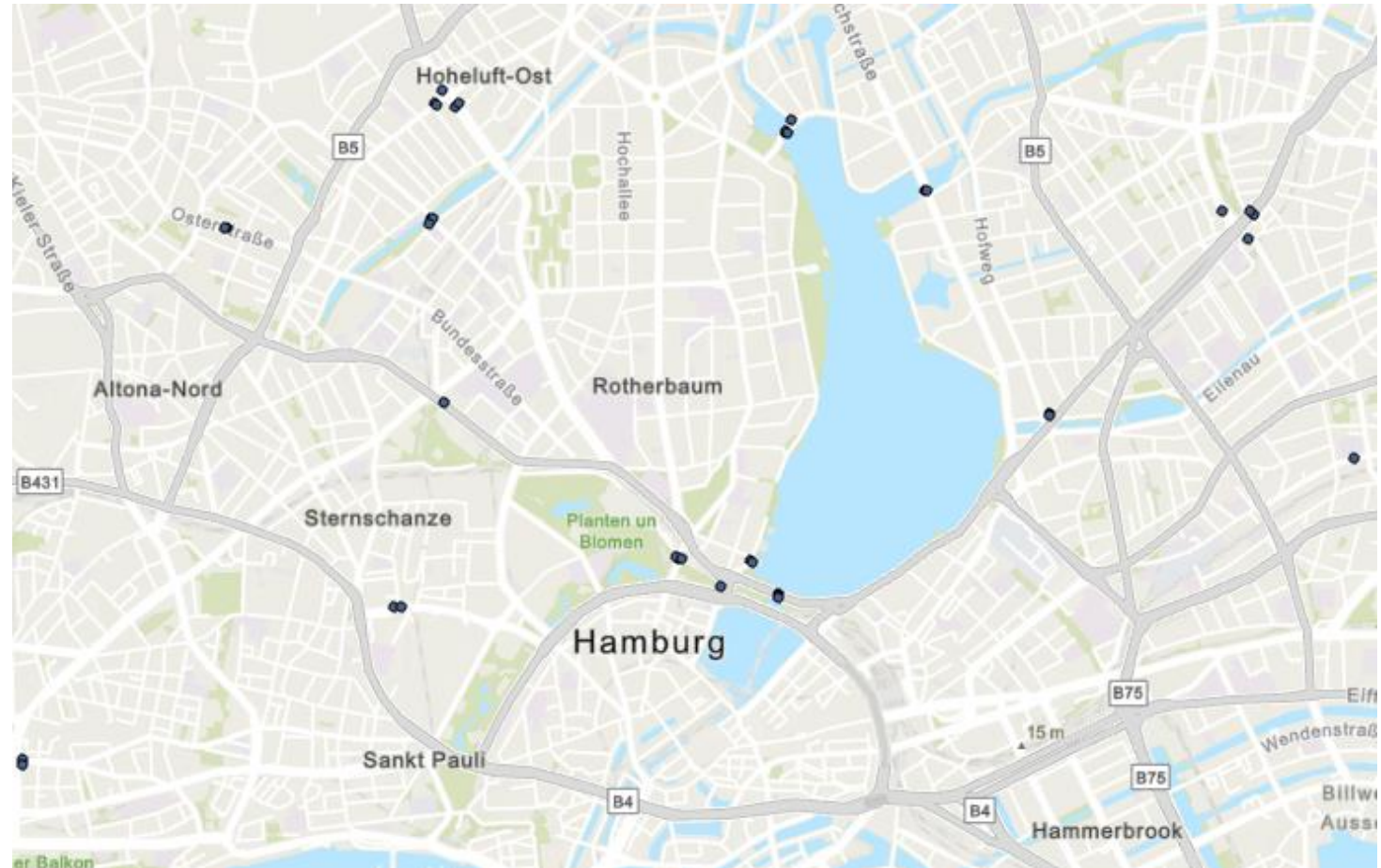
Fahrradzählstationen

/data/generated/wfs/bike_count.geojson

Geometrie Punkt

Quelle HH_WFS_Harazaen

Anzahl 133



Fahrradzählstationen

/data/generated/wfs/bike_count.geojson

| | |
|--------------------------------------|---|
| <i>radfahrende_seit_jahresbeginn</i> | Anzahl Radfahrender seit Beginn des Jahres |
| <i>tageslinie</i> | Anzahl Radfahrender pro Stunde pro Tag (konfigurierbar) |
| <i>radfahrende_insgesamt</i> | Anzahl Radfahrender seit Beginn der Zählung |
| <i>radfahrende_vorjahr</i> | Anzahl Radfahrender im Vorjahr |
| <i>max_radfahrende_woche_jahr</i> | Woche, mit den meisten Radfahrenden in diesem Jahr (+ Anzahl) |
| <i>typ</i> | Typ der Zählstation |
| <i>link_download</i> | Link zu Download dieser Daten |
| <i>max_radfahrende_monat_jahr</i> | Monat, mit den meisten Radfahrenden in diesem Jahr (+ Anzahl) |
| <i>in_betrieb_seit</i> | Datum der Inbetriebnahme der Station |
| <i>max_radfahrende_tag_jahr</i> | Tag, mit den meisten Radfahrenden in diesem Jahr (+ Anzahl) |
| <i>jahrgangslinie</i> | Anzahl Radfahrender pro Woche in diesem Jahr |
| <i>name</i> | Name der Station |

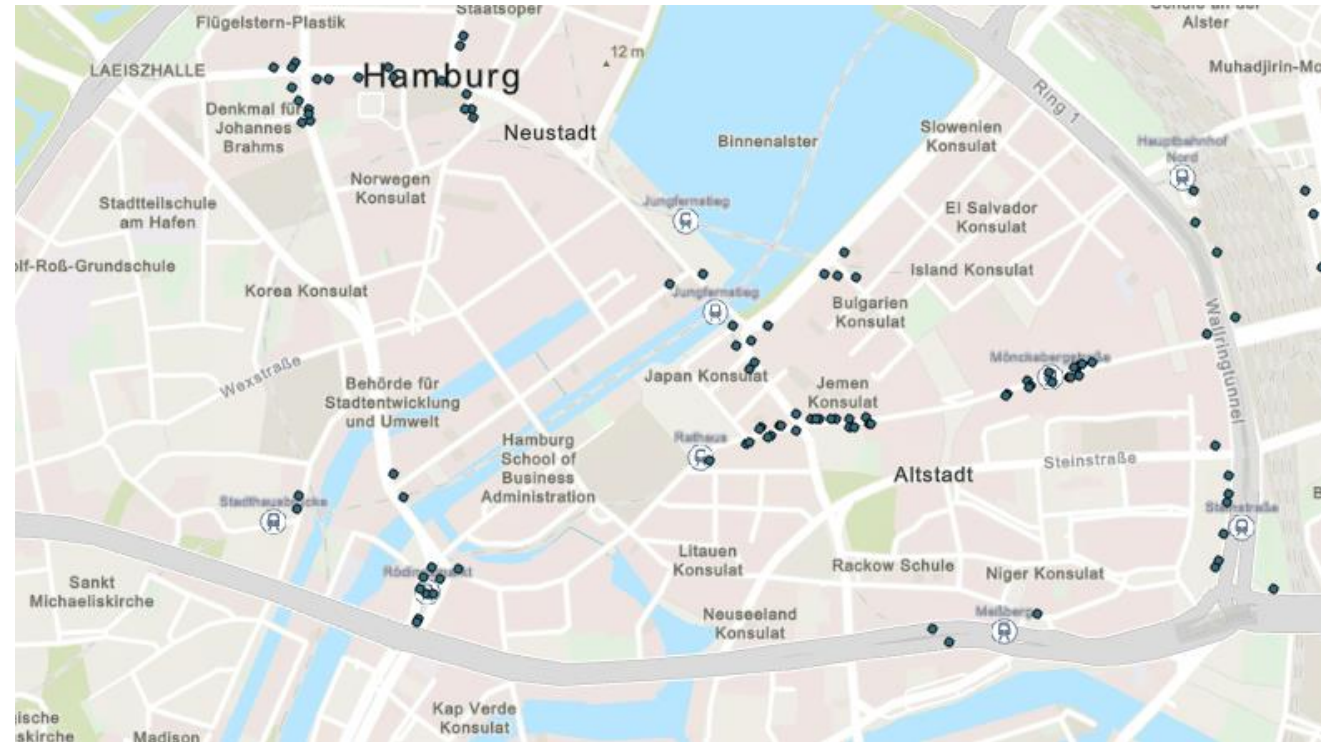
Bike & Ride Stationen

/data/generated/wfs/bike_and_ride.geojson

Geometrie Punkt

Quelle HH_WFS_Bike_und_Ride

Anzahl ~ 1.300



Bike & Ride Stationen

/data/generated/wfs/bike_and_ride.geojson

| haltestelle | Haltstelle die zu der B+R Anlage gehört |
|---|---|
| <i>symbol</i> | ? |
| <i>adresse</i> | Adresse der B+R Anlage |
| <i>davon_nicht_ueberdachte_stellplaetze</i> | Anzahl nicht überdachter Stellplätze |
| <i>davon_gesicherte_mietstellplaetze</i> | Anzahl gesicherter Mietstellplätze |
| <i>stand</i> | Letzte Aktualisierung |
| <i>email</i> | Email-Adresse für Kontakt |
| <i>davon_ueberdachte_stellplaetze</i> | Anzahl überdachter Stellplätze |
| <i>schliessfaecher</i> | Anzahl Schließfächer |
| <i>internet</i> | Internet Seite der B+R |

Fahrradluftstationen/Fahrradladestationen

/data/generated/wfs/bike_air_station.geojson

Geometrie Punkt

Quelle HH_WFS_Fahrradluftstationen

Anzahl 49

Attribute

betreiber

Betreiber der Station

name

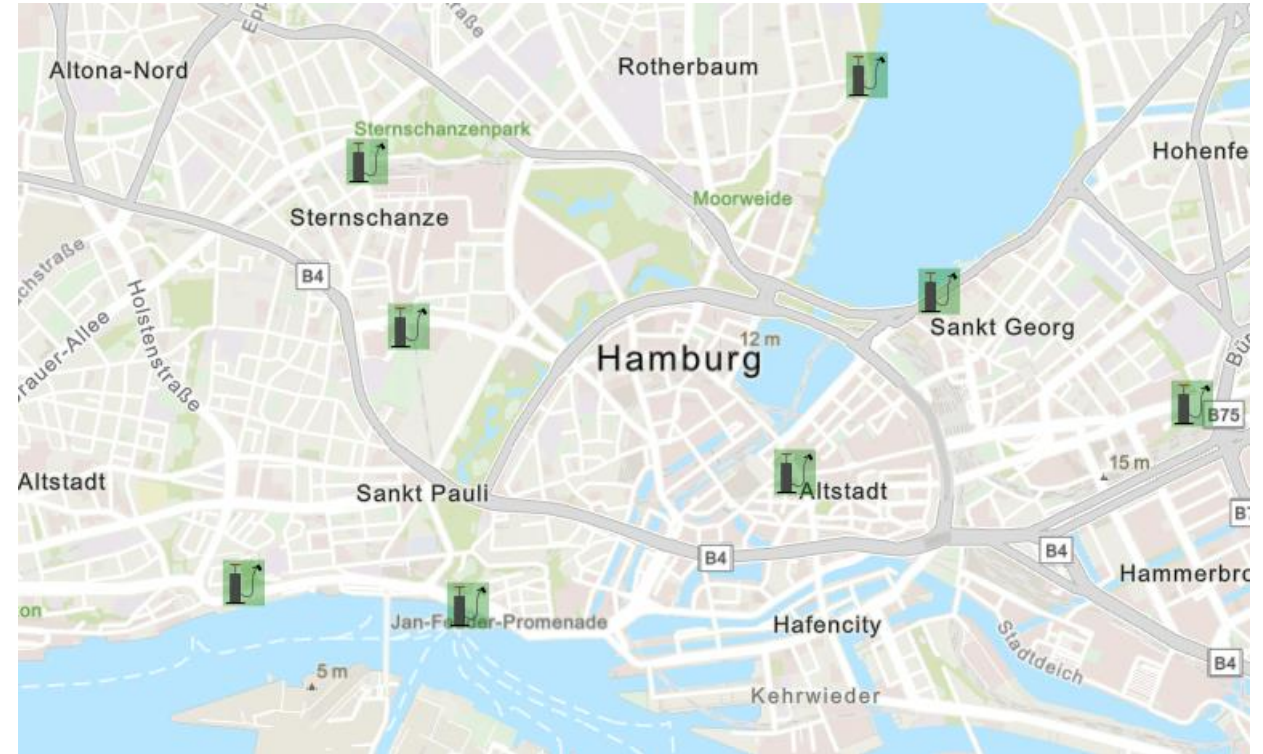
Name der Station

kontakt

Email / Telefonnummern

anmerkungen

Art der Pumpe, Anzahl Verfügbarer Ladestationen



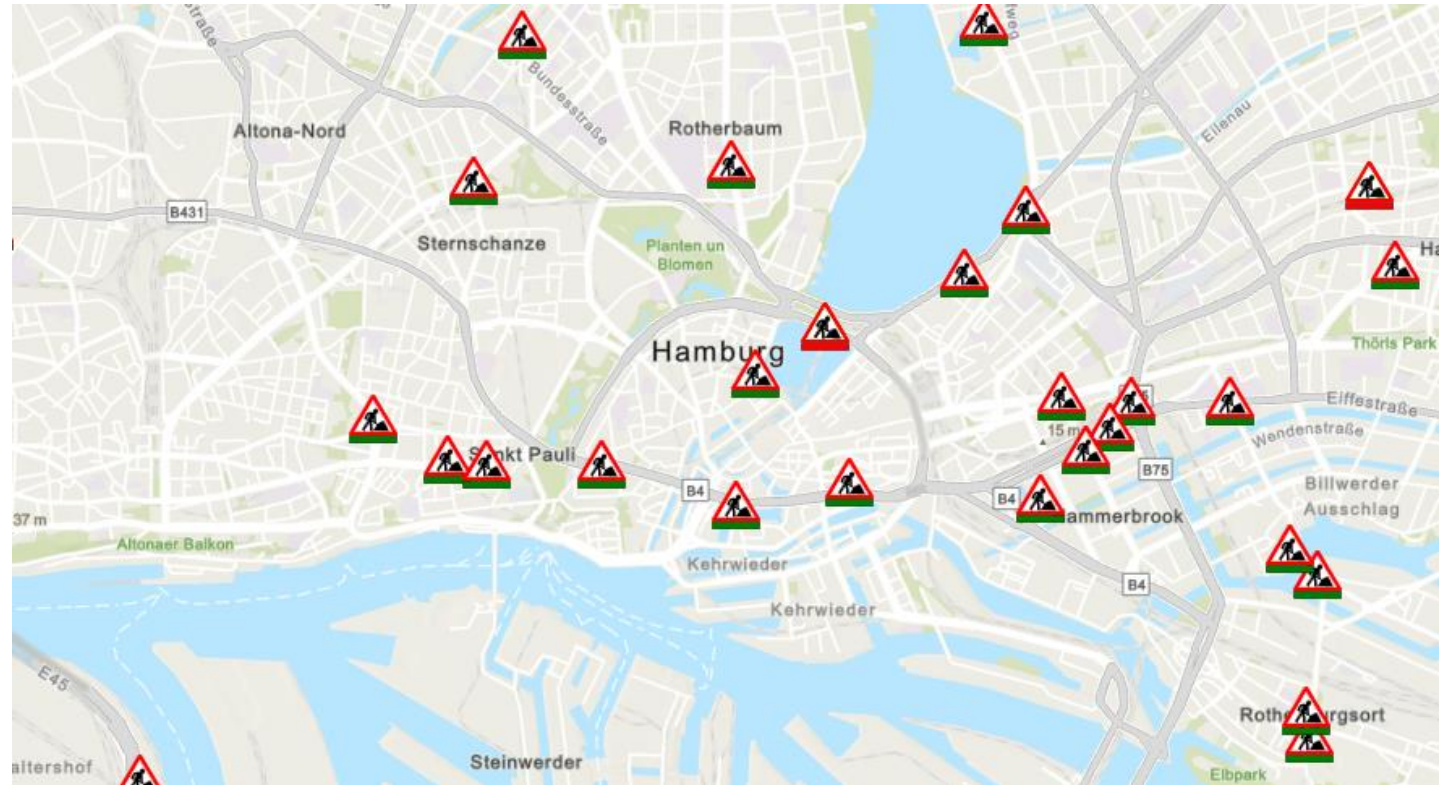
Baustellen

/data/generated/wfs/construction_sides.geojson

Geometrie Punkt

Quelle HH_WFS_Baustellen

Anzahl Variabel



Baustellen

/data/generated/wfs/construction_sides.geojson

```
<element name="gml_identifizier" minOccurs="0" type="string"/>
<element name="gml_identifizier_attr_codespace" minOccurs="0" type="string"/>
<element name="titel" minOccurs="0" type="string"/>
<element name="organisation" minOccurs="0" type="string"/>
<element name="anlass" minOccurs="0" type="string"/>
<element name="umfang" minOccurs="0" type="string"/>
<element name="baubeginn" minOccurs="0" type="string"/>
<element name="bauende" minOccurs="0" type="string"/>
<element name="letzteaktualisierung" minOccurs="0" type="dateTime"/>
<element name="istzugangeingeschraenkt" minOccurs="0" type="boolean"/>
<element name="einschraenkungsbegrueundung" minOccurs="0" type="string"/>
<element name="koordinate_attr_nilreason" minOccurs="0" type="string"/>
<element name="koordinate_attr_gml_remoteschema" minOccurs="0" type="string"/>
<element name="koordinate_attr_owns" minOccurs="0" type="boolean"/>
<element name="iststoerung" minOccurs="0" type="boolean"/>
<element name="isthotspot" minOccurs="0" type="boolean"/>
<element name="istfreigegeben" minOccurs="0" type="boolean"/>
<element name="mehrwert" minOccurs="0" type="string"/>
<element name="istoeopnveingeschraenkt" minOccurs="0" type="boolean"/>
<element name="oevnveinschraenkungen" minOccurs="0" type="string"/>
<element name="hatinternetlink" minOccurs="0" type="boolean"/>
<element name="internetlink" minOccurs="0" type="string"/>
<element name="hatumleitungsbeschreibung" minOccurs="0" type="boolean"/>
<element name="umleitungsbeschreibung" minOccurs="0" type="string"/>
<element name="istparkraumeingeschraenkt" minOccurs="0" type="boolean"/>
<element name="parkraumeinschraenkung" minOccurs="0" type="string"/>
<element name="einsatzkraefte" minOccurs="0" type="string"/>
<element name="tns_koordinate_value" minOccurs="0" type="gml:GeometryPropertyType"/>
```

Verkehr

```
/data/generated/wfs/traffic.geojson
```

Geometrie Polylinie

Quelle HH_WFS_Verkehrslage

Anzahl Variabel

Attribute

Strassenklasse

Art der Straße

Zeitstempel_utc

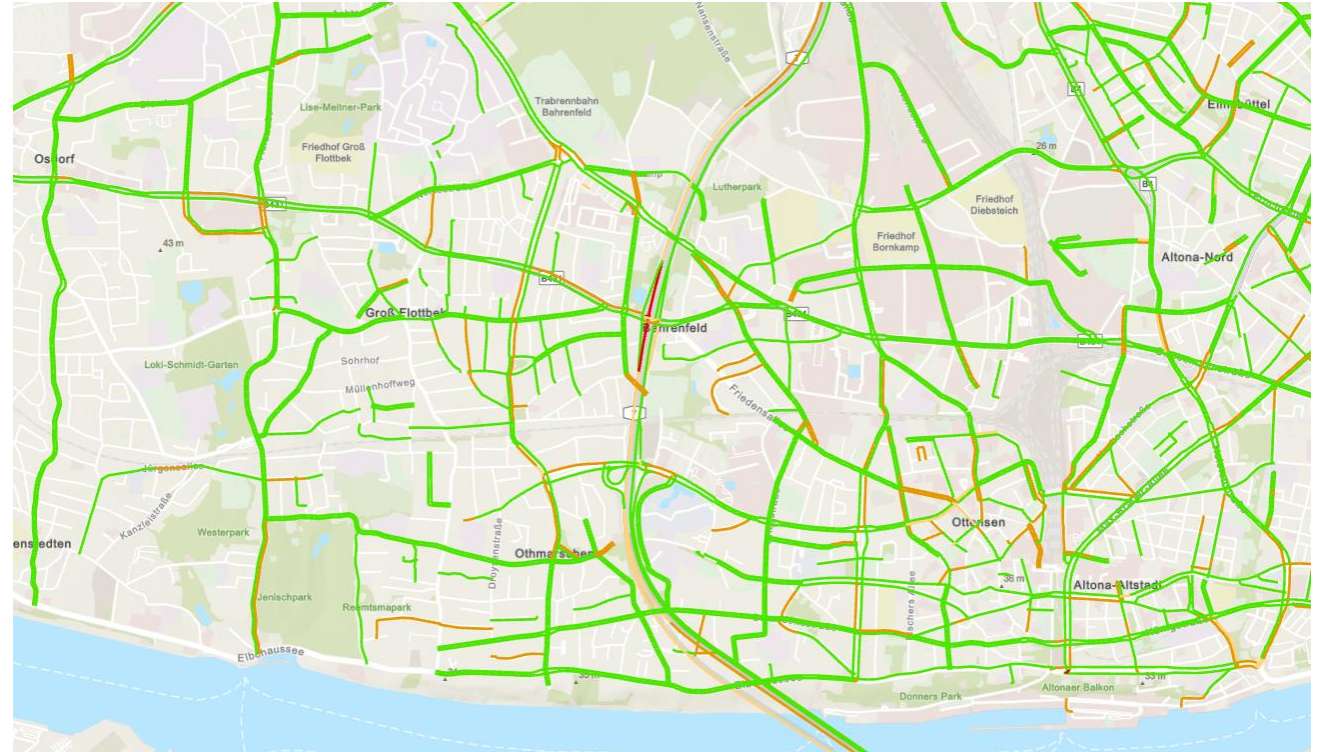
Zeitstempel in UTC

Zustandsklasse

Auslastung der Straße

zeitstempel

Zeitstempel in CET/CEST



Multi-Criteria Decision Analysis

Ermittlung von Unfallschwerpunkten in Hamburg

Kriterien/Daten

Suchbereich Hamburg

Kriterien

- Anzahl der Unfälle
- Schwere der Unfälle
- Dichte der Unfälle
- Stärke des Verkehrs

Unfälle

- Quelle <https://unfallatlas.statistikportal.de/>
- Beteiligte Verkehrsteilnehmer, Schwere des Unfalls

Straßen

- Quelle OpenStreetMap
- Art der Straße (viel befahrene Straßen)

Verwaltungsgrenzen Hamburg

- <http://opendatalab.de/projects/geojson-utilities/>

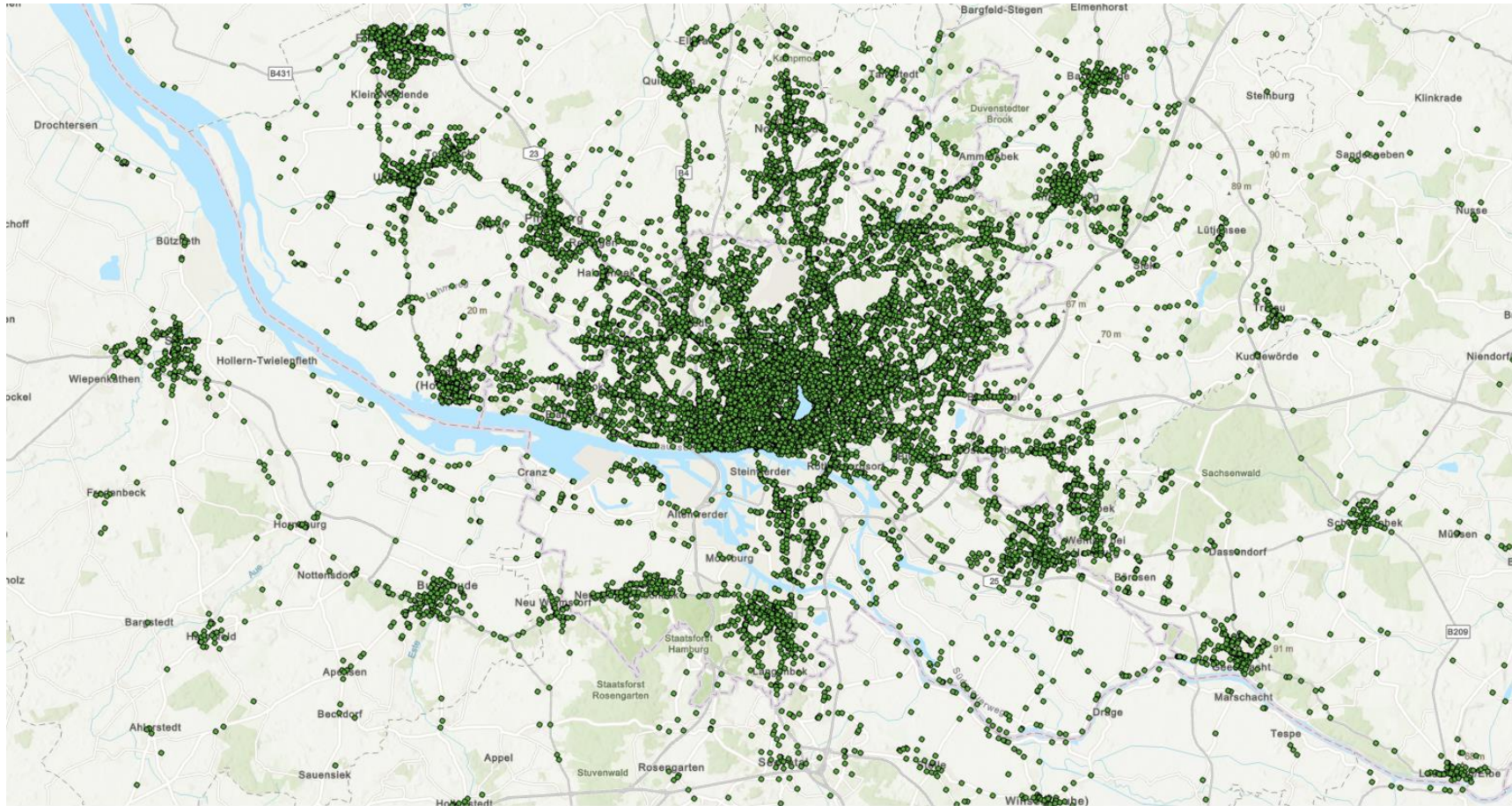
Vorgehensweise

Skript `generate_accident_black_spots.py`

- Führt bei Bedarf Download aus (get_accidents Funktion) (~ 15-20 Minuten)
- Alle Scores und Variablen sind konfigurierbar
- Ausführungszeit Berechnung ~2 Minuten
- Beliebig erweiterbar auf andere Gebiete (benötigt neue Straßendaten & Boundary)

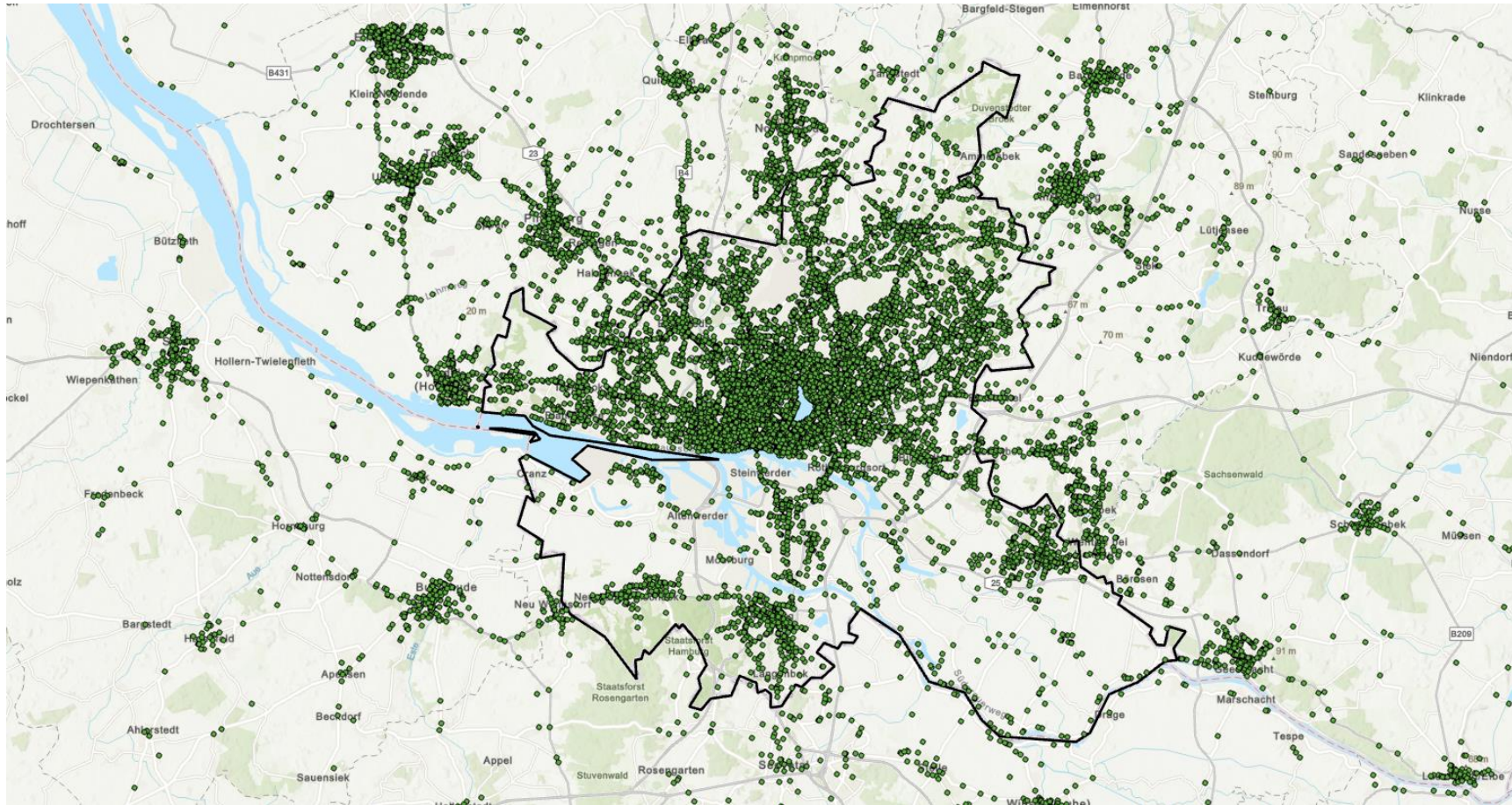
Vorgehensweise

Unfalldaten herunterladen



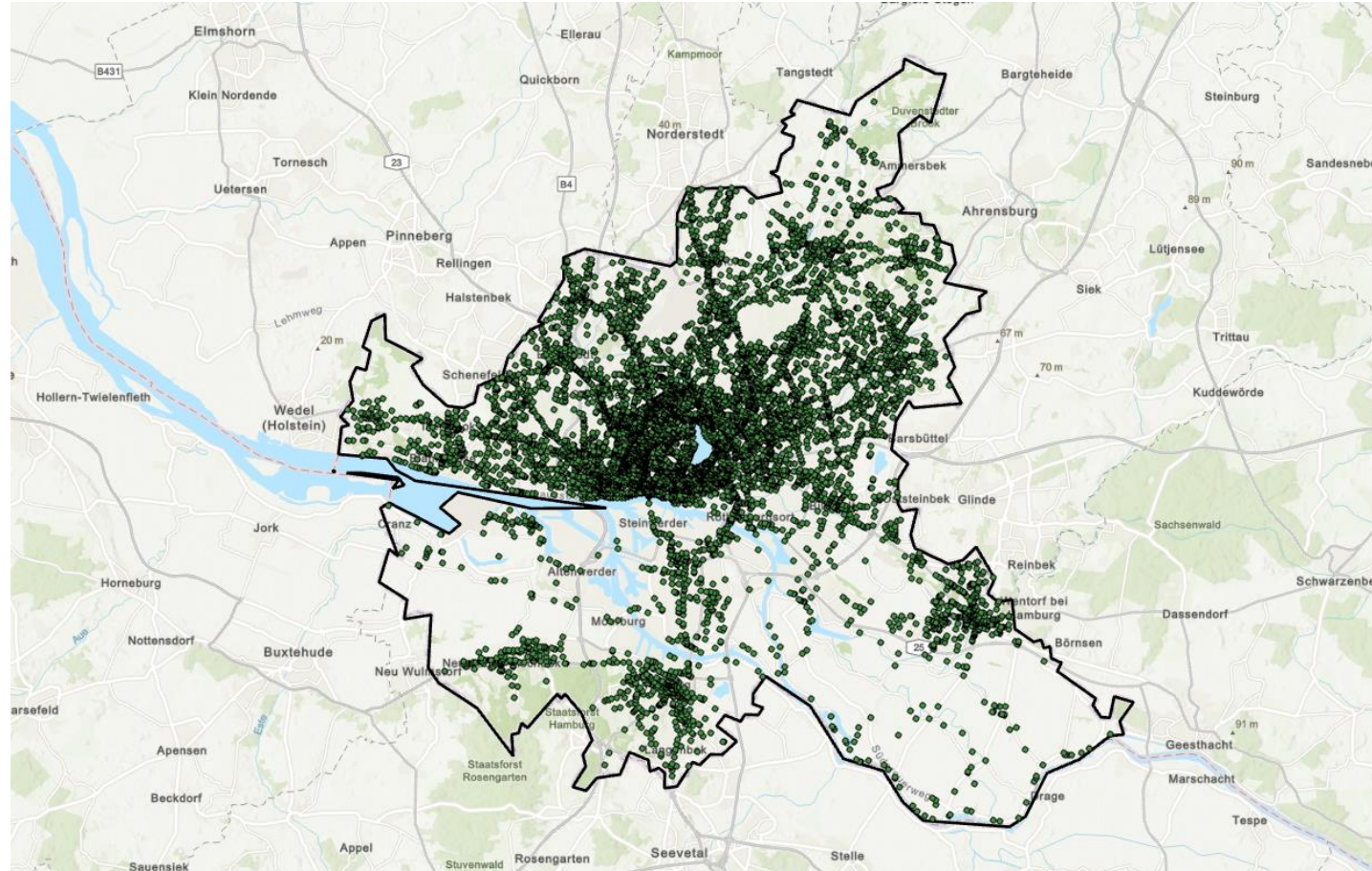
Vorgehensweise

Grenzen von Hamburg



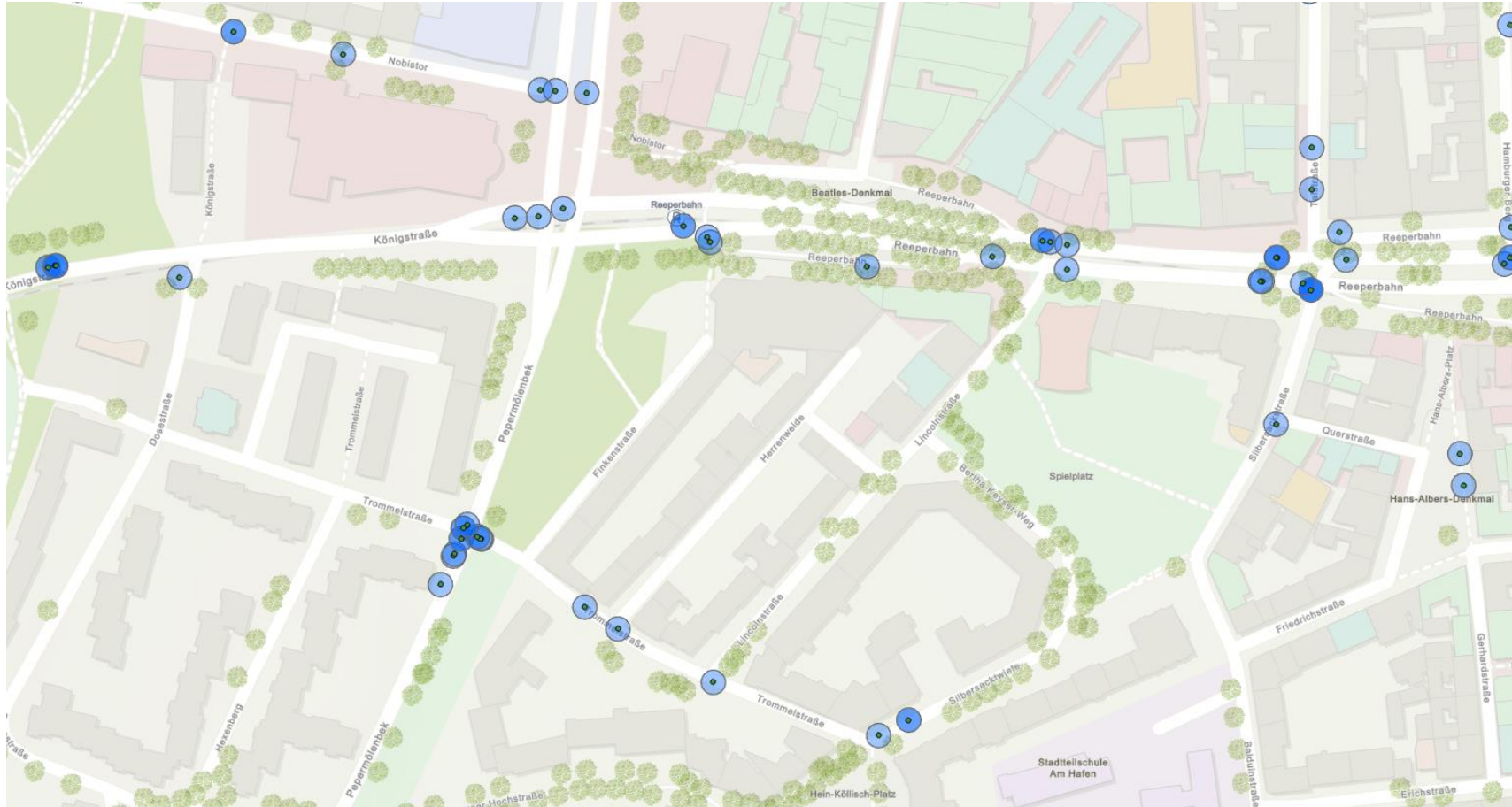
Vorgehensweise

Grenzen von Hamburg



Vorgehensweise

Puffern von Punktdaten zur „Gruppierung“



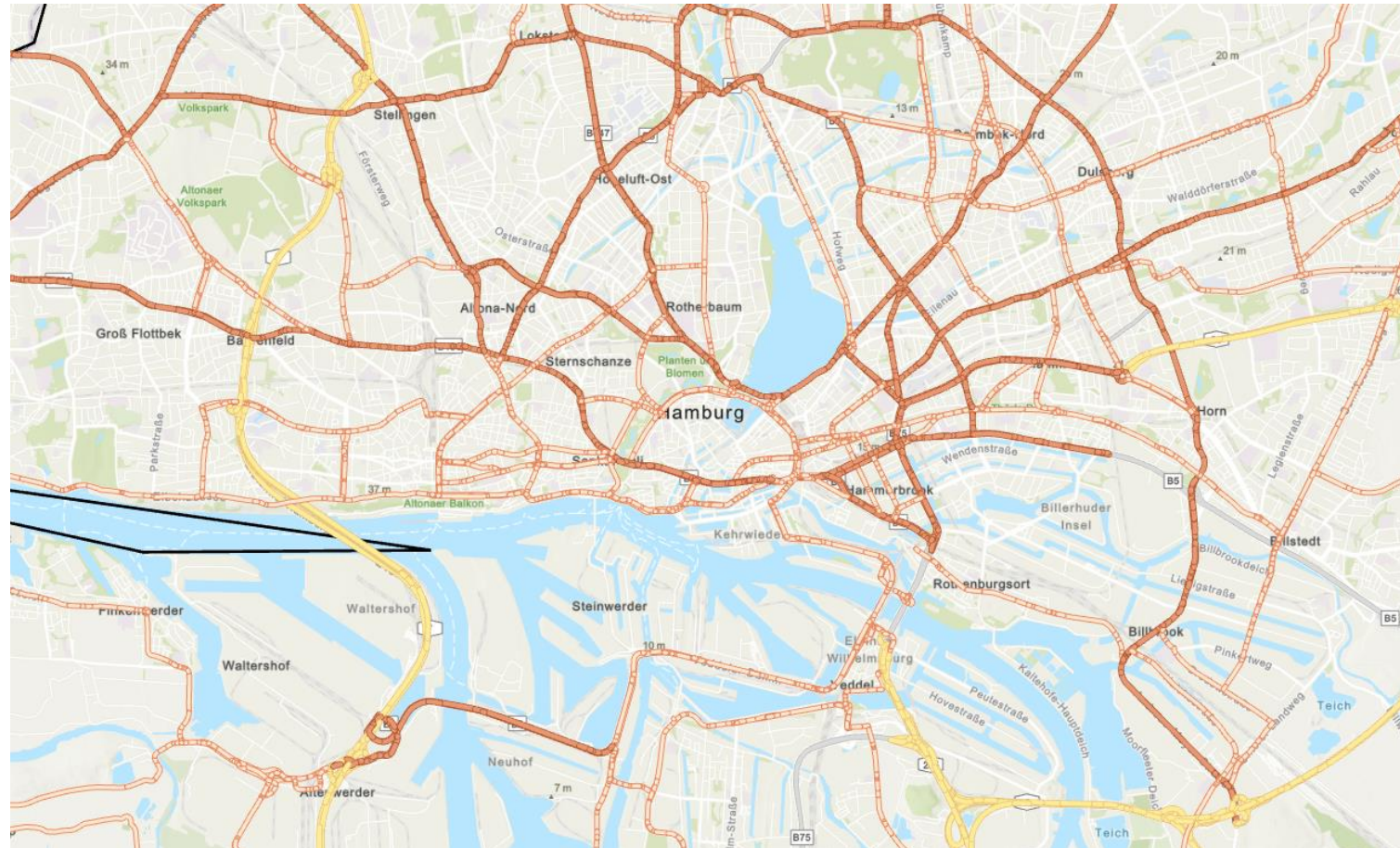
Vorgehensweise

Zusammenführen überlappender Puffer (Dissolve)



Vorgehensweise

Straßendaten (gefiltert nach Hauptverkehrsadern)



Vorgehensweise

Score Berechnung Straßen

- Verschneiden von Straßen mit Unfallpufferobjekte
 - Motorway ... 1
 - Primary road ... 1.2
 - Secondary road ... 1.5
 - Keine ... 2

score_road

- Maximum Wert der verschnittenen Straßen

Vorgehensweise

Score Berechnung Dichte

- Referenzwert: Radius Buffer * 3.1415

score_area

- Referenzwert – (Fläche Unfallpuffer / Anzahl Unfall)

Vorgehensweise

Score Berechnung Verletzlichkeit

- Verschneiden von Unfalldaten mit Unfallpufferobjekte
 - Leichtverletzte ... 1
 - Schwerverletzte ... 2
 - Tote ... 3

score_vulnerabilty

- $\text{Summe Schwere Unfall} / \text{Anzahl Unfälle}$

Ergebnis

total_score

- $\text{score_road} * \text{score_vulnerabilty} * \text{score_area}$

Score kann nicht in mit anderen Scores verglichen werden:

- Extremer Unfallschwerpunkt würde alle anderen validen Unfallschwerpunkte „klein“ wirken lassen
- Fester Grenzwert

Koordinatensystem WGS84

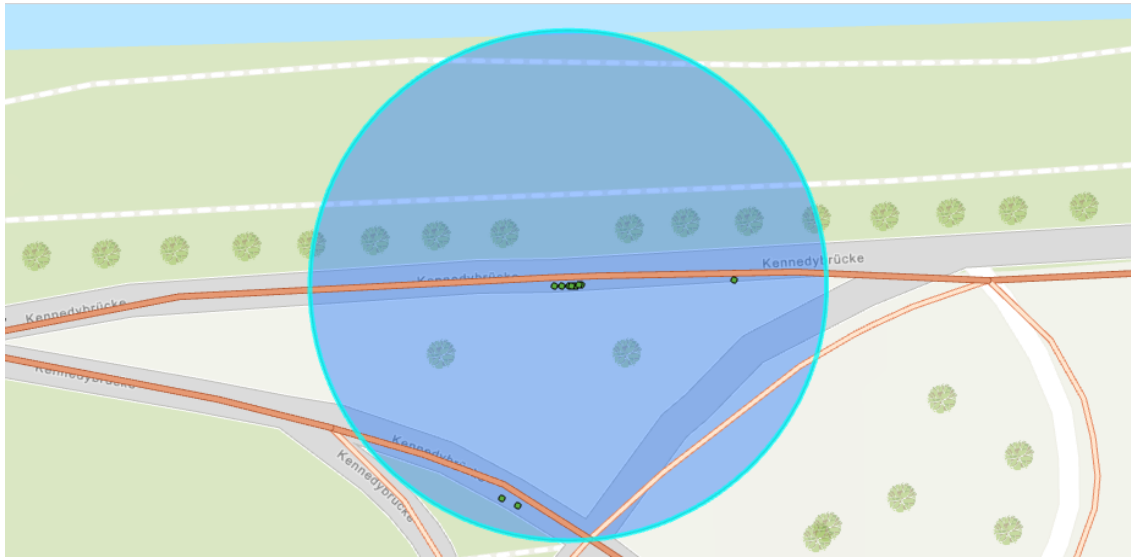
= 6 Unfallschwerpunkte



Diskussion

Reicht pure Anzahl der Unfälle nicht aus?

11 Unfälle
Unfallschwerpunkt



16 Unfälle
Kein Unfallschwerpunkt



Einbindung in die App

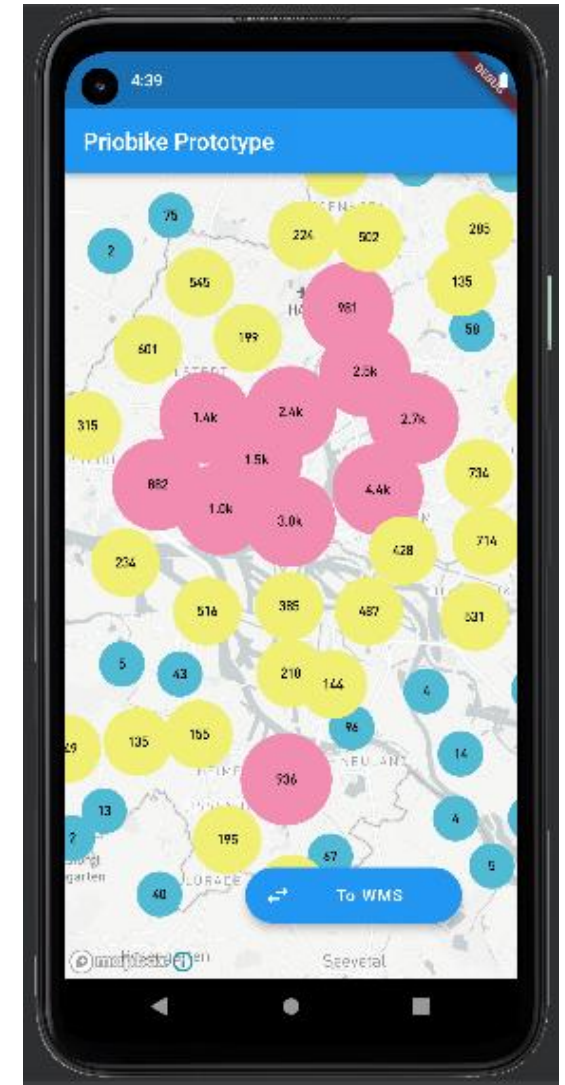
Mapbox GL, PostGIS, GeoServer

Einbindung von GeoJSON

Flutter Mapbox GL per Layer

MapboxMapController

- `addSource(name_source, GeojsonSourceProperties)`
 - Fügt Datenquelle hinzu
 - Link zu GeoJSON Datei
 - Konfiguration von Clustern (`cluster`, `clusterMaxZoom`, `clusterRadius`, ...)
- `addLayer(name_source, name_layer, LayerProperties)`
 - Baut aus Datenquelle Layer
 - Fügt diesen zum Controller hinzu
 - Konfiguration des Layers



Einbindung von GeoJSON

Flutter Mapbox GL per Symbol

Ungeeignet für große Anzahl von Features

MapboxMapController

- `addSymbol(SymbolOptions(geometry, iconImage))`



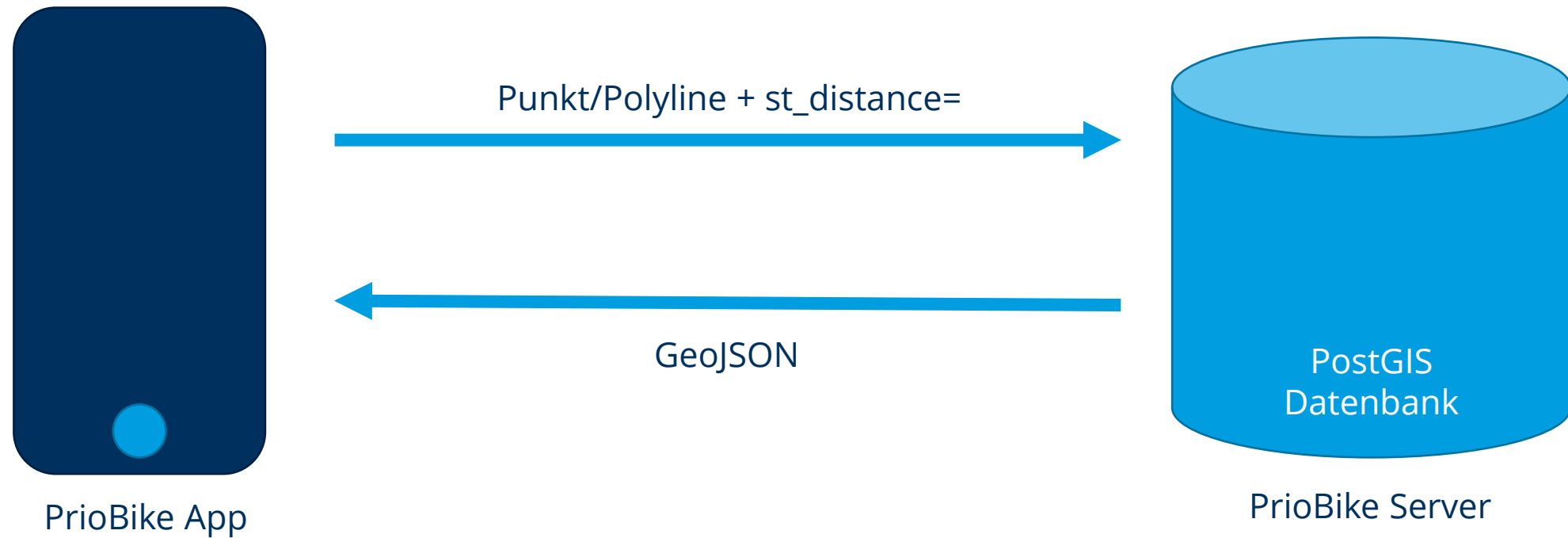
Einbindung von WMS

MapboxController

- addSource(name_source, RasterSourceProperties/ImageSourceProperties)
 - url, tiles, ...
 - Anfrage an WMS mit BoundingBox
- addLayer(name_source, name_layer, RasterLayerProperties)
 - Baut aus Datenquelle Layer
 - Fügt diesen zum Controller hinzu
 - Konfiguration des Layers

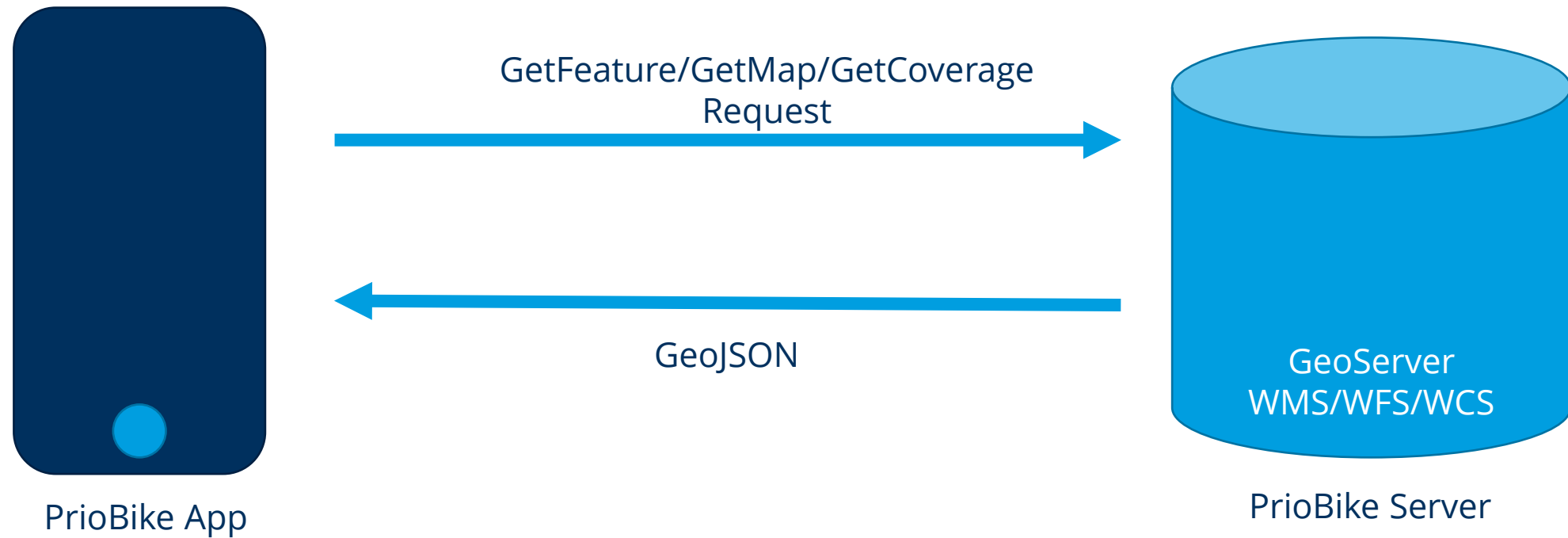
Kommunikation mit Server

PostGIS



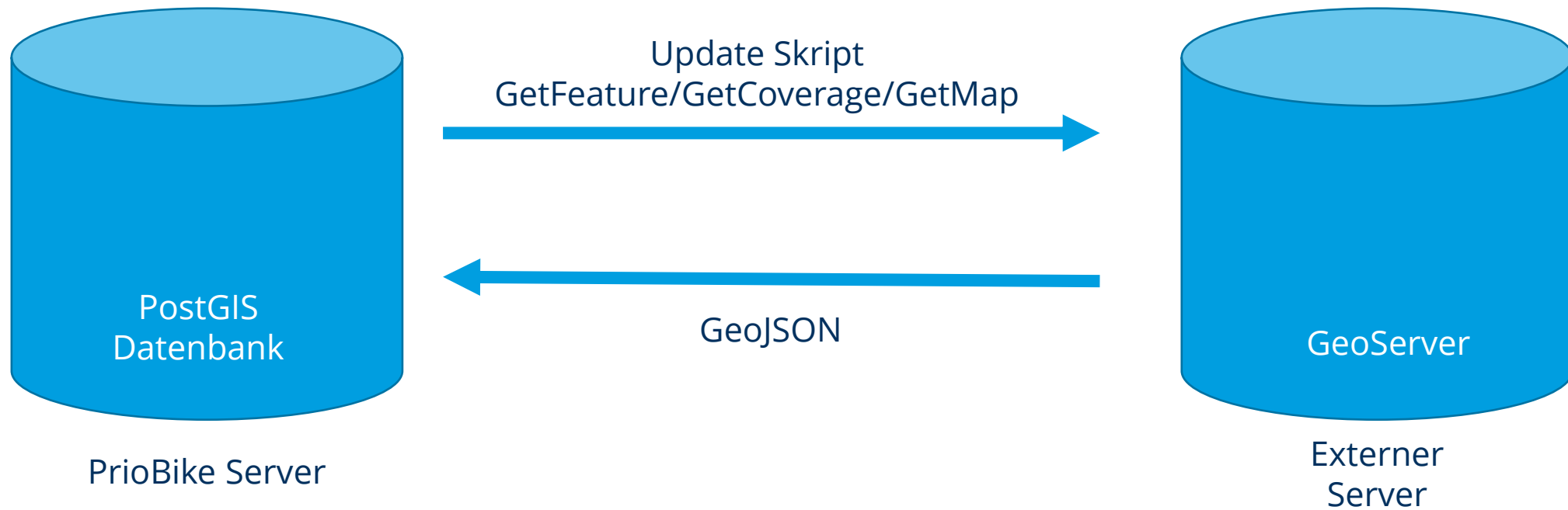
Kommunikation mit Server

Geo-Webservice



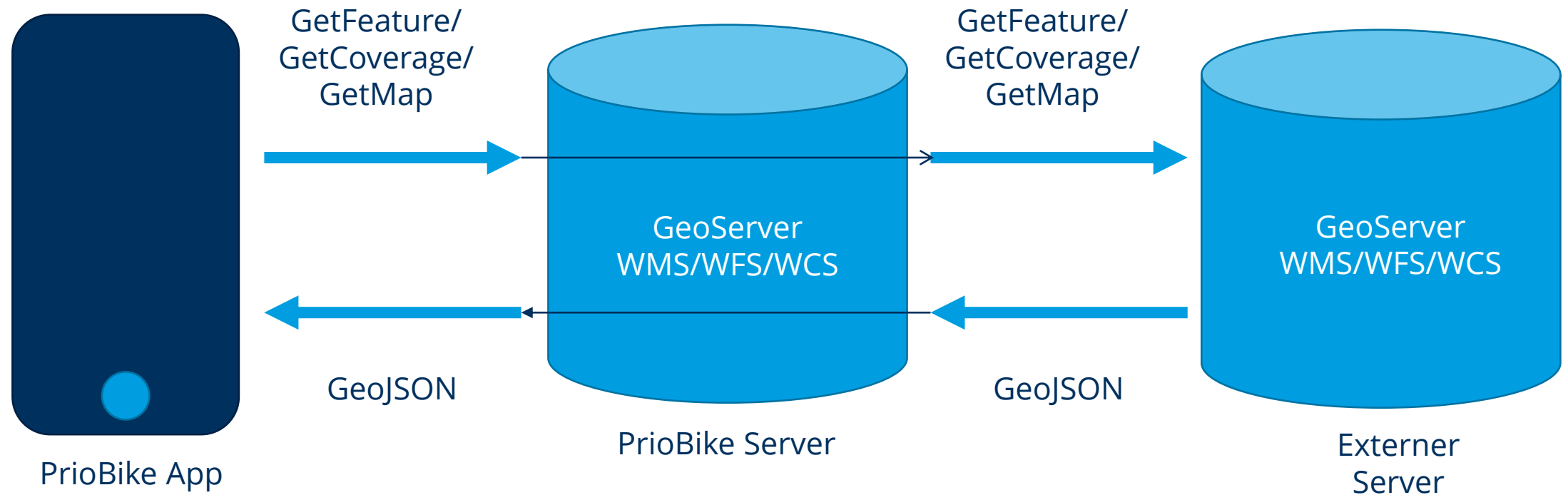
Update der Daten

Variante 1



Update der Daten

Variante 2



Fragen?

Quellen

<https://geoportal-hamburg.de/verkehrsportal/#>

<https://geoportal-hamburg.de/geo-online/>

<https://transparenz.hamburg.de/>

<https://unfallatlas.statistikportal.de/>

<https://download.geofabrik.de/europe/germany/hamburg.html>

<https://maps.dwd.de/geoserver/wms?SERVICE=WMS&VERSION=1.3.0&REQUEST=GetCapabilities>

<http://opendatalab.de/projects/geojson-utilities/>