

Overpass API - die Datenbank zur OpenStreetMap

Mit der Overpass API die Datenvielfalt aus der OpenStreetMap kitzeln

Linuxwochen Wien, 3. Mai 2019 - plepe.at/356

Stephan Bösch-Plepelits - skunk@openstreetmap.at - plepe.at

Social media:

- Twitter: twitter.com/plepe
- Mastodon: [@plepe@en.osm.town](https://en.osm.town/@plepe)
- Jabber: skunk@jabber.at

Struktur

- OpenStreetMap
- Overpass API
- Anwendungsbeispiele

OpenStreetMap - was ist das?

- <https://www.openstreetmap.org/>
- Eine von der Community erstellte Karte der Welt
(c) OpenStreetMap-Mitwirkende
- Open Data: ODbL (Rohdaten) bzw. CC-BY-SA 2.0 (Kacheln)
- Schirmherrschaft: OpenStreetMap Foundation

OpenStreetMap - wie funktioniert das?

3 verschiedene Objekttypen:

- Node: ID; Tags; Lat/Lon Koordinaten
- Way: ID; Tags; Liste von Nodes
- Relation: ID; Tags; Liste von Nodes, Ways und Relationen (mit Rolle)

Tags sind Liste von Key/Value Paaren, z.b.:

- amenity=restaurant, cuisine=kebap;pizza, name=Kebaphaus
- highway=primary, name=Dresdner Straße, maxspeed=50, oneway=yes

https://wiki.openstreetmap.org/wiki/Map_Features

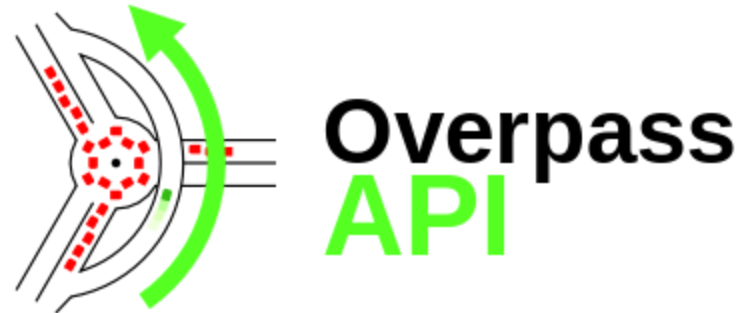
Was ist die Overpass API?

Eine effiziente Datenbank um Daten aus der OpenStreetMap auszulesen.

->

https://wiki.openstreetmap.org/wiki/Overpass_API

- Abfragesprache: Overpass QL
- Free Software (GNU AGPL 3.0)
- Public Server: z.B. overpass-api.de
- Output: JSON, XML, CSV



Beispiel für eine Overpass Query

```
[out:json][bbox:48.21,16.37,48.22,16.38];  
node[amenity=cafe];  
out body geom;
```

```
{ "version": 0.6,  
  "generator": "Overpass API 0.7.55",  
  "elements": [  
    { "type": "node", "id": 319735103,  
      "lat": 48.2079420, "lon": 16.3698350,  
      "tags": {  
        "amenity": "cafe",  
        "name": "Café Hawelka",  
        "name:zh": "哈維卡咖啡",  
        "website": "http://www.hawelka.at",  
        "wikipedia": "de:Café Hawelka"  
      }  
    }, ...  
  ]  
}
```

Overpass Turbo

Frontend für Overpass API: <https://overpass-turbo.eu>

The screenshot displays the Overpass Turbo web interface. On the left, a code editor shows the following query:

```
1 [out:json][bbox:{{bbox}}];  
2  
3 nwr[amenity=cafe];  
4  
5 out body geom;
```

The main area shows a map of Vienna. A popup window for Node 319735103 is open, displaying the following information:

Node 319735103

Tags:

- addr:city=Wien
- addr:country=AT
- addr:housenumber=6
- addr:postcode=1010
- addr:street=Dorotheergasse
- amenity=cafe
- name=Café Hawelka
- name:zh=哈維卡咖啡
- official_name=Café Leopold Hawelka
- website=<http://www.hawelka.at>
- wheelchair=limited
- wheelchair:description=Zwei kleine Stufen-Absätze beim Eingang, kein Rollstuhl-WC
- wikidata=Q494438
- wikipedia=de:Café Hawelka

Coordinates:

48.207942 / 16.369835 (lat/lon)

At the bottom right, a status bar indicates: geladen – Nodes: 25, Ways: 0, Relations: 0; angezeigt – POIs: 25, Linien: 0, Polygone: 0.

© OpenStreetMap-Mitwirkende, <https://osm.org/copyright>

CC-BY-SA 4.0 Stephan Bösch-Plepelits - plepe.at/356

Overpass Turbo - Features

- "Wizard" für das schnelle Erstellen von Abfragen
- "Export" in verschiedene Formate: GeoJSON, GPX, KML, OSM
- "Teilen" Permalinks erstellen
- Daten-Ansicht

Overpass QL (1/6)

Overpass Query Language. Jede Query besteht aus mehreren Teilen:

1. Einstellungen
2. Selektierung
3. Ausgabe

Nach einer Ausgabe kann wieder eine weitere Query folgen.

Im folgenden werden nur die wichtigsten Konstrukte erklärt, eine vollständige Anleitung gibt es auf

https://wiki.openstreetmap.org/wiki/Overpass_API/Overpass_QL

Overpass QL (2/6) - Einstellungen

```
[out:json] vs. [out:xml] vs. [out:csv] // Format  
[bbox:48.2,16.2,48.4,16.4] // Region (default: Welt)  
[date:"2018-12-24T19:00:00Z"] // Historische Daten  
[diff:] and [adiff:] // Änderungen (verwende: [out:xml])
```

Overpass QL (3/6) - Selektierung

Typ:

```
node; way; relation;  
nwr /* alles */, area /* flächen um punkt */
```

Selektiere Tags:

```
nwr[amenity=cafe]; // Tag hat Wert  
nwr[amenity]; // Tag hat irgendeinen Wert  
nwr[amenity~"^(bar|cafe)$"]; // Regular Expression  
nwr[~"name"~"straße$"]; // Regular Expression  
node(id:1234,1235); // Nach ID
```

Bedeutung der Tags:

https://wiki.openstreetmap.org/wiki/DE:Map_Features

Overpass QL (4/6) - Ausgabe

```
out;
```

Detailgrad:

- `ids` : nur IDs
- `meta` : inkludiere Meta-Daten (timestamp, user, ...)

Geo-Informationen:

- `geom` : Volle Geometrie, inkl. Members
- `bb` : Nur bounding box
- `center` : Nur Centroid

Overpass QL (5/6) - Weitere Konstrukte

Union:

```
( node[amenity=cafe]; node[amenity=bar]; );
```

Members:

```
way[highway]; out; >; out;
```

Sets:

```
node[amenity=cafe]->.a; node.a[cuisine]; .a out;
```

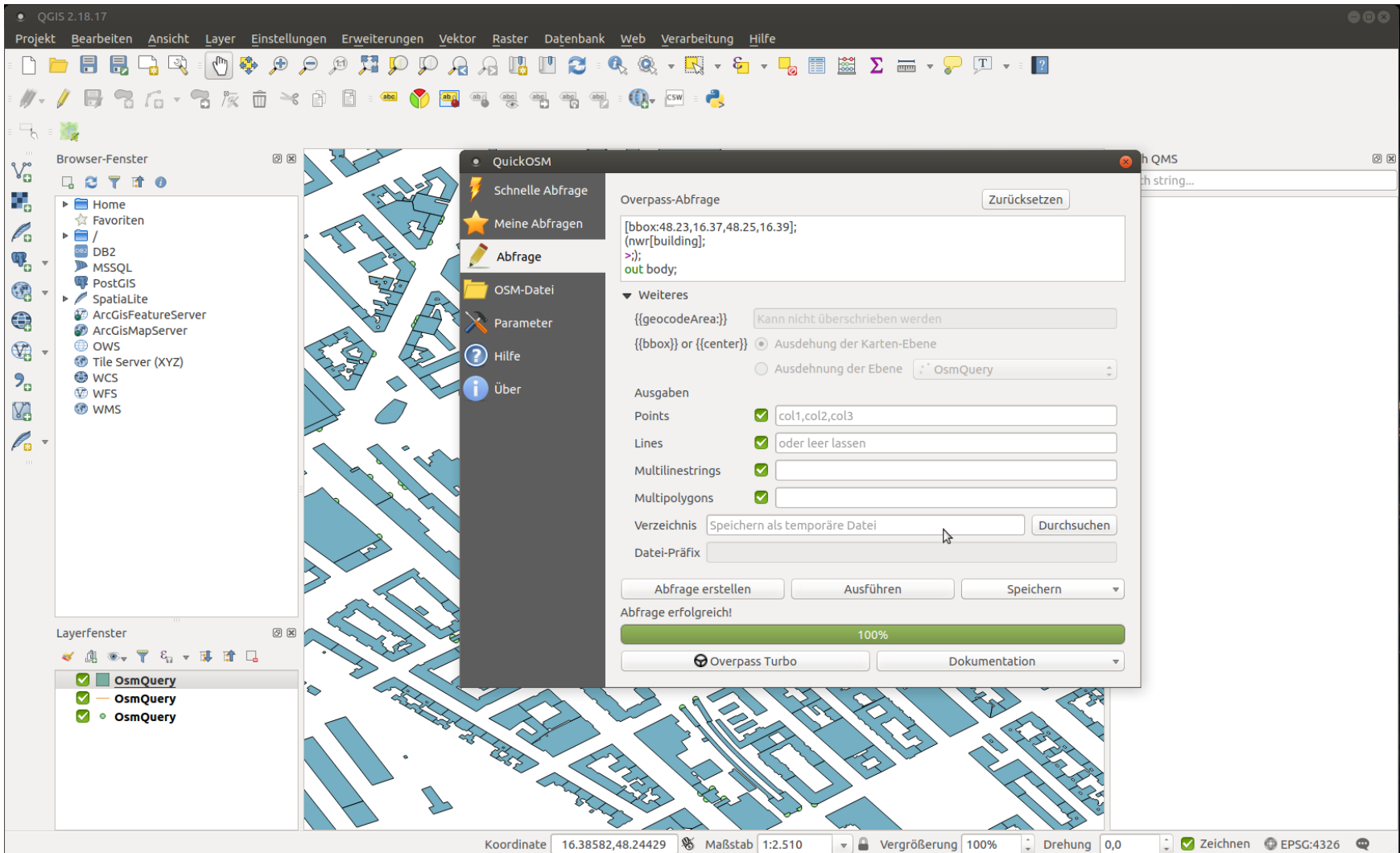
Overpass QL (6/6) - komplexes Beispiel

Alle Kinos die nicht mehr als 100m von einer Bus- oder Bimstation in Wien entfernt sind.

```
area[name="Wien"];  
(  
  node(area)[highway=bus_stop];  
  node(area)[railway=tram_stop];  
);  
nwr(around:100)[amenity=cinema];  
out;
```

Overpass API & QGIS

- Plugin QuickOSM: <https://github.com/3liz/QuickOSM>



Exportiere Daten kompatibel mit anderen Tools

Mit der folgenden Abfrage sind die Daten mit anderen OSM-Tools (z.B. JOSM, Osmosis) kompatibel.

```
[out:xml][bbox:{{bbox}}];  
  
way[highway]; // Or any other query  
  
out meta;  
>;  
out meta;
```


Abfragen mit CURL direkt in eine Datei schreiben

```
curl \
  -X POST \
  -d "[out:json];node[place=continent];out;" \
  https://overpass-api.de/api/interpreter \
  > export.json
```

Webseiten mit eingebetteter Karte und Overpass Overlay erstellen

LeafletJS + verschiedene

- <https://github.com/plepe/overpass-layer>
- <https://github.com/GuillaumeAmat/leaflet-overpass-layer>

OpenStreetBrowser

- <https://www.openstreetbrowser.org>



OpenStreet
Browser
4.6



Freizeit, Sport und Einkauf

Dienste

Kultur und Religion

Kultur



- Johann Strauß Denkmal *Kunstwerke*
 - Vogeltränke-Brunnen *Springbrunnen*
 - Donauweibchen-Brunnen *Springbrunnen*
 - Befreiung der Quelle-Brunnen *Springbrunnen*
 - Wettersäule *Kunstwerke*
 - Blumenuhr *Kunstwerke*
- weitere Ergebnisse anzeigen

Geschichte

Denkmäler und Monumente



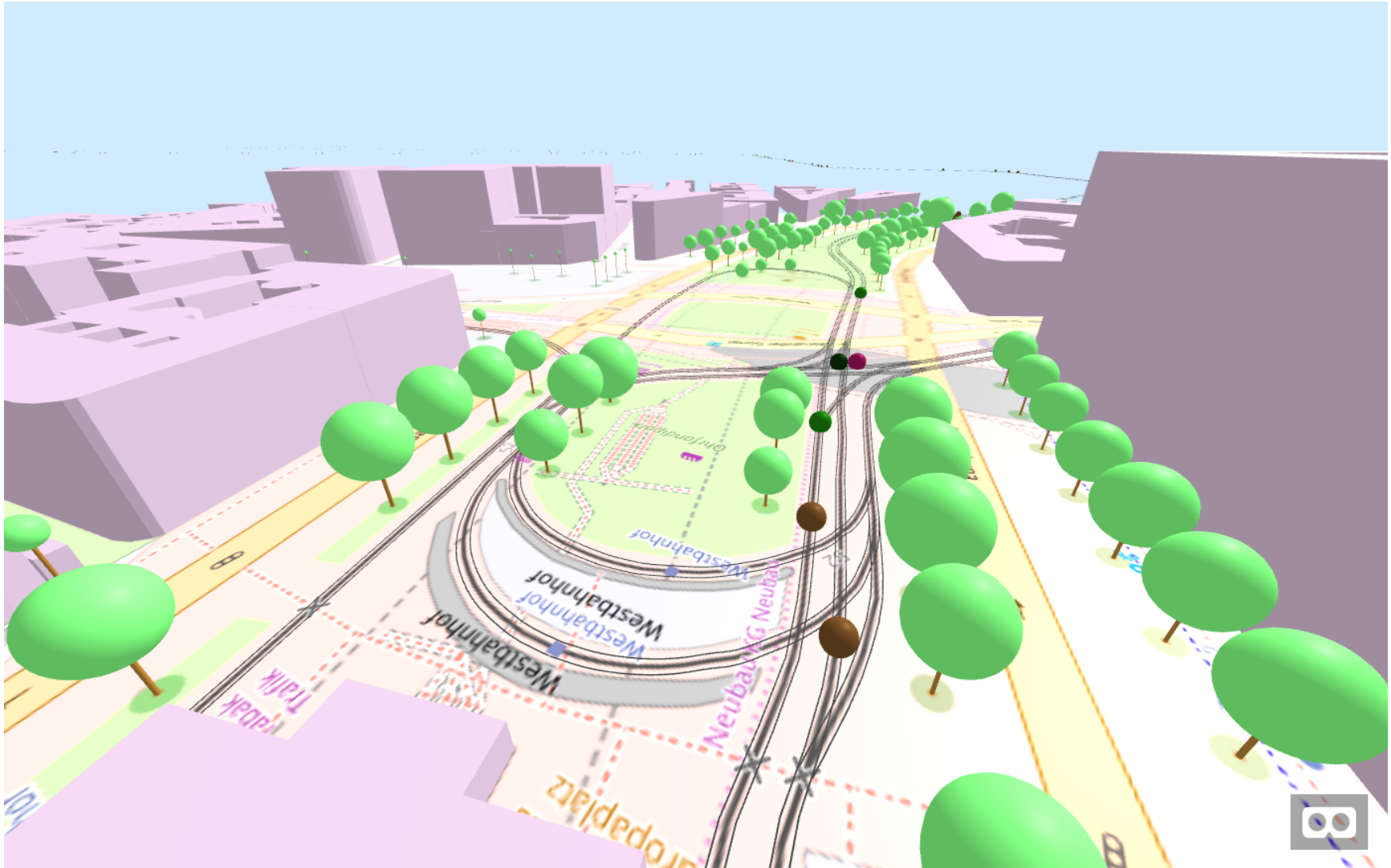
- Franz Schubert *Gedenkstätten*
- Franz Lehár *Gedenkstätten (Bust)*
- Friedrich von Amerling *Gedenkstätten (Bust)*
- Namenlos *Gedenkstätten (Commemorative)*

Code | Optionen | Permalink



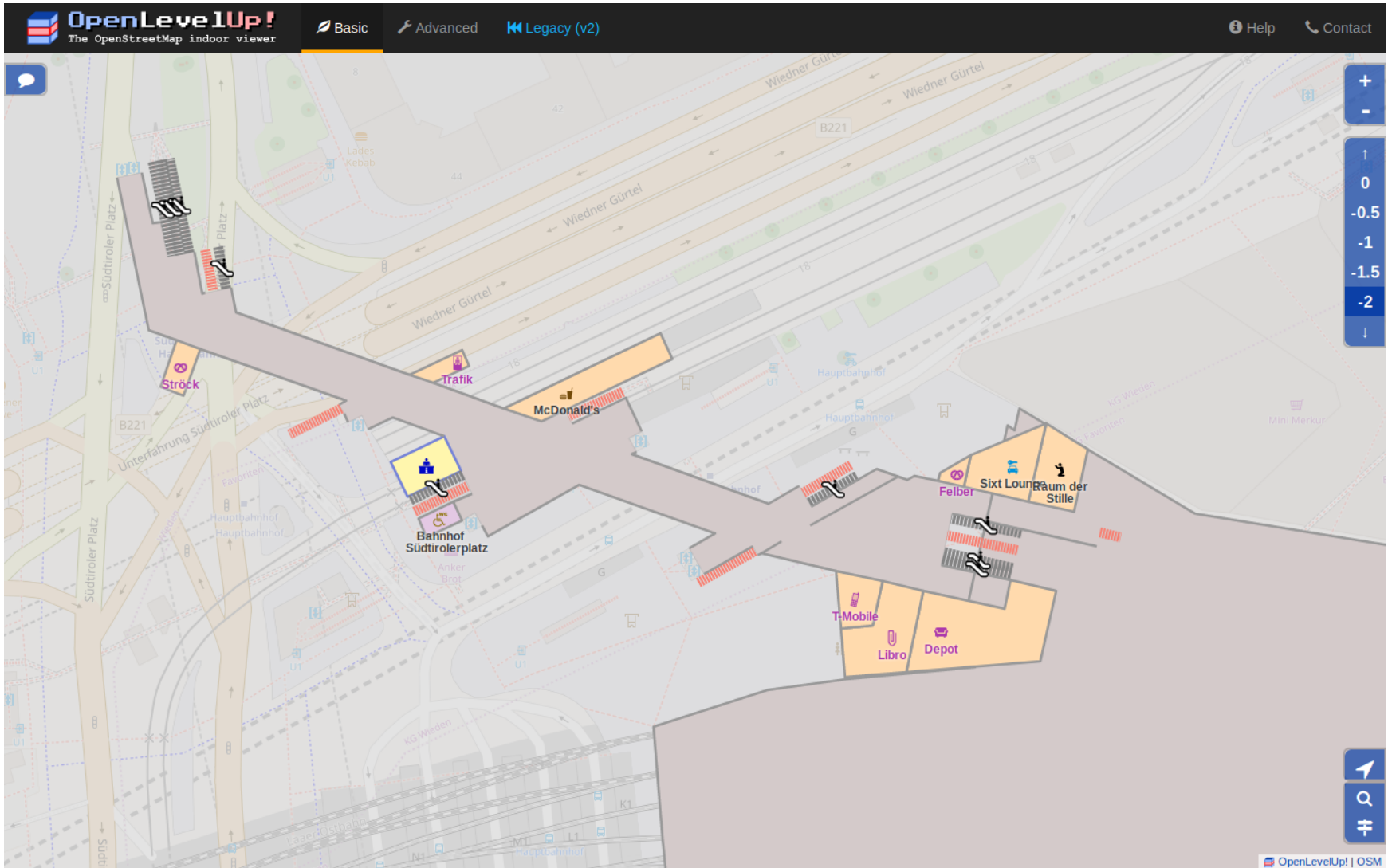
VR Map

- <https://vrmap.kairo.at> - <https://vrmap.plepe.at>



OpenLevelUp!

- <https://openlevelup.net/>



Happy mapping!

Danke für Eure Aufmerksamkeit!

Fragen?

References

- Abbildungen © OpenStreetMap-Mitwirkende
- Marp (Markdown presentation editor)