

Einführung Overpass

Stuttgarter Stammtisch

Übersicht

- Was ist Overpass?
- wichtige Links
- getting started
- Wizzard
- erste Abfrage
- Weiterentwicklung der Abfrage
- die Bounding Box
- Verknüpfungen
- Ausgabe
- Kommandos
- Warnungen
- Weitere Themen

Was ist Overpass ?

- **Overpass turbo** ist ein webbasierendes Datensammelwerkzeug für OpenStreetMap. Die Webadresse ist <http://overpass-turbo.eu>.
- Es läuft mit jeder [Overpass-API-Abfrage](#) und zeigt die Resultate in einer interaktiven Karte.

wichtige Links

- Main developers:
 - [Turbo: Martin Raifer](#) (mail: martin@raifer.tech)
<https://wiki.openstreetmap.org/wiki/User:Tyr>)
 - [API: Roland Olbricht](#) (mail: roland.olbricht@gmx.de)
<https://wiki.openstreetmap.org/wiki/User:Roland.olbricht>
- <http://overpass-turbo.eu/> „GUI“
- [https://wiki.openstreetmap.org/wiki/DE:Overpass turbo](https://wiki.openstreetmap.org/wiki/DE:Overpass_turbo) Wiki
- <http://overpass-api.de/index.html> Doku
- [https://wiki.openstreetmap.org/wiki/Overpass API/Overpass QL](https://wiki.openstreetmap.org/wiki/Overpass_API/Overpass_QL) Query Lang
- [https://wiki.openstreetmap.org/wiki/Overpass API/Overpass API by Example](https://wiki.openstreetmap.org/wiki/Overpass_API/Overpass_API_by_Example)
- [https://wiki.openstreetmap.org/wiki/Overpass API](https://wiki.openstreetmap.org/wiki/Overpass_API) API Dok

getting started

<http://overpass-turbo.eu/>

Befehle

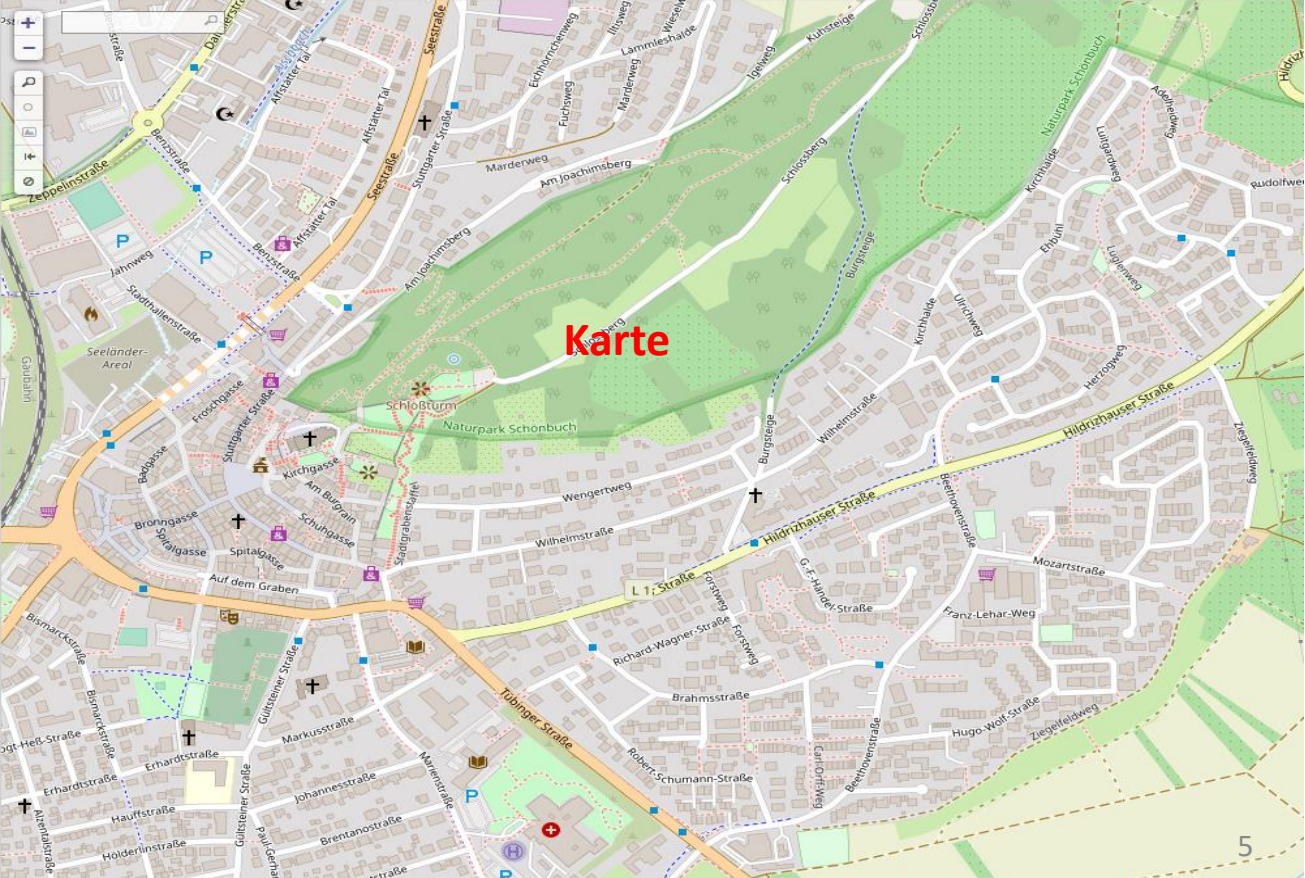
AusführenTeilenExportWizardSpeichernLadenAusloggenEinstellungenHilfeoverpass turbo

1
2
3
4
5
6
7
8

```
area[admin_level=8]
[name="Herrenberg"]->.searchArea;
(
  // rel[type=route][route=bicycle]
  (area.searchArea);
  rel[type=route][route=hiking]
  (area.searchArea);
);
out body;
>; out skel qt;
```

KarteDaten

Karte



Kommandos

5

Wizzard

Abfrage-Assistent

amenity=restaurant in Böblingen

Dieser **Wizzard** hilft dabei selbst Overpass Abfragen zu erstellen. Hier sind einige Nutzungsbeispiele:

- Drinking Water
- highway=* and type:way
- tourism=museum in Vienna

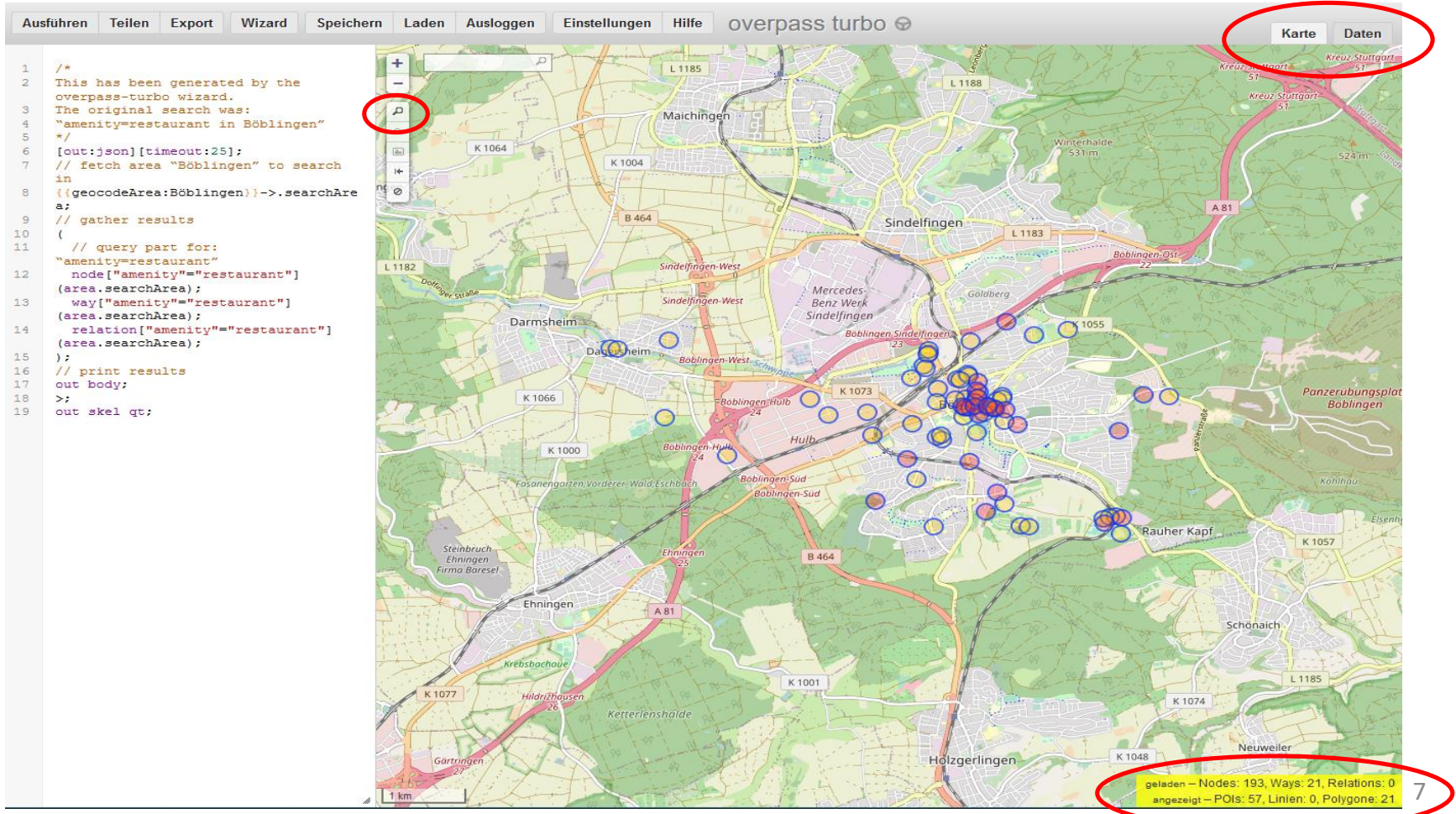
Abfrage erstellen

Abfrage erstellen und ausführen

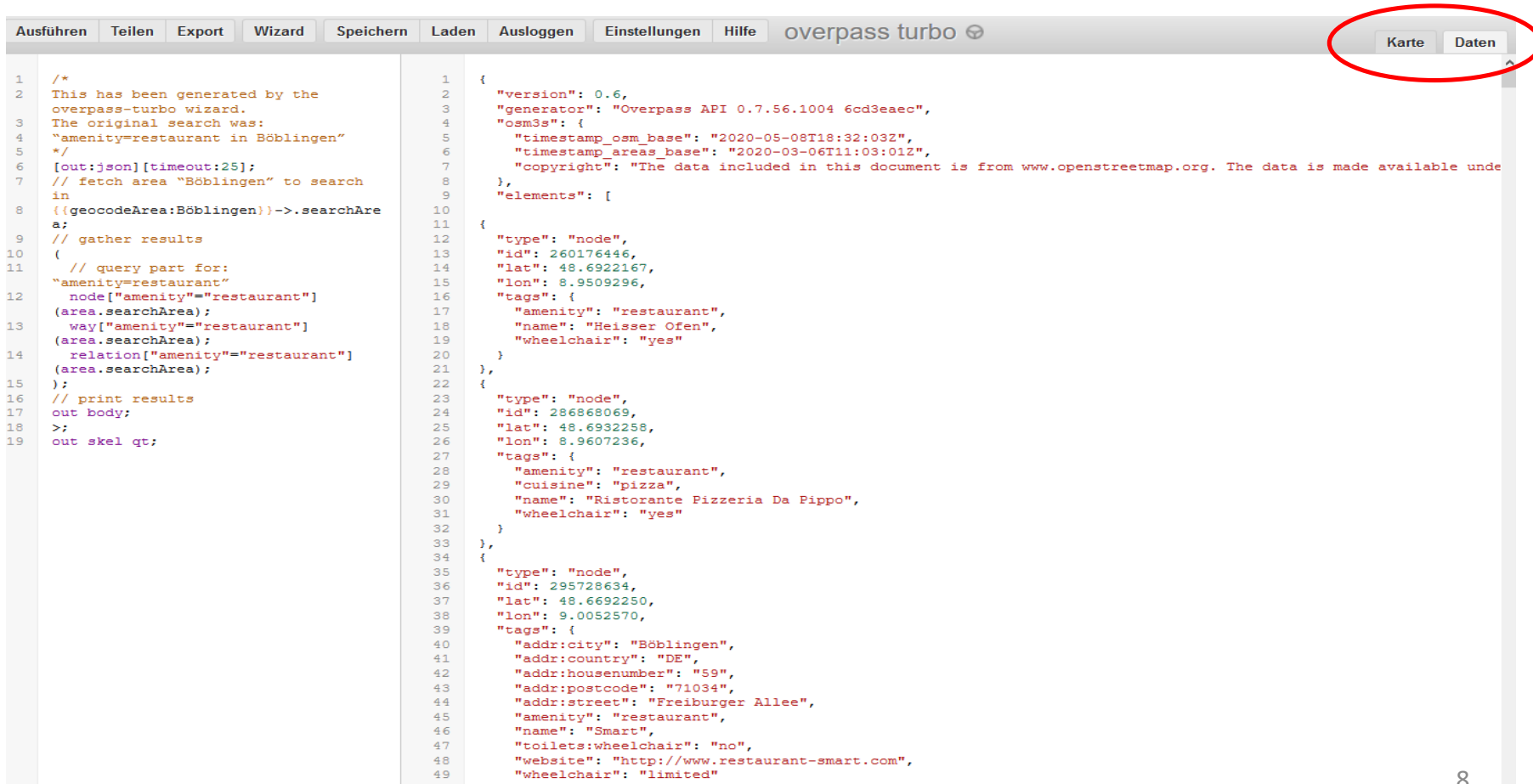
Abbrechen

```
1. /*
2. This has been generated by the overpass-turbo wizard.
3. The original search was:
4. "amenity=restaurant in Böblingen"
5. */
6. [out:json][timeout:25];
7. // fetch area "Böblingen" to search in
8. {{geocodeArea:Böblingen}}->.searchArea;
9. // gather results
10. (
11. // query part for: "amenity=restaurant"
12. node["amenity"]="restaurant"({area.searchArea});
13. way["amenity"]="restaurant"({area.searchArea});
14. relation["amenity"]="restaurant"({area.searchArea});
15. );
16. // print results
17. out body;
18. >;
19. out skel qt;
```


erste Abfrage



erste Abfrage



The screenshot shows the Overpass Turbo web interface. The top navigation bar includes buttons for 'Ausführen', 'Teilen', 'Export', 'Wizard', 'Speichern', 'Laden', 'Ausloggen', 'Einstellungen', and 'Hilfe'. The 'overpass turbo' logo is on the right, with 'Karte' and 'Daten' buttons circled in red. The main area is split into two panes. The left pane contains a query script, and the right pane displays the resulting JSON data.

```
1  /*
2  This has been generated by the
3  overpass-turbo wizard.
4  The original search was:
5  "amenity=restaurant in Böblingen"
6  */
7  [out:json][timeout:25];
8  // fetch area "Böblingen" to search
9  in
10  {{geocodeArea:Böblingen}}->.searchArea;
11  // gather results
12  (
13  // query part for:
14  "amenity=restaurant"
15  node["amenity"="restaurant"]
16  (area.searchArea);
17  way["amenity"="restaurant"]
18  (area.searchArea);
19  relation["amenity"="restaurant"]
20  (area.searchArea);
21  );
22  // print results
23  out body;
24  >;
25  out skel qt;
```

```
1  {
2    "version": 0.6,
3    "generator": "Overpass API 0.7.56.1004 6cd3eaec",
4    "osm3s": {
5      "timestamp_osm_base": "2020-05-08T18:32:03Z",
6      "timestamp_areas_base": "2020-03-06T11:03:01Z",
7      "copyright": "The data included in this document is from www.openstreetmap.org. The data is made available unde
8    },
9    "elements": [
10
11    {
12      "type": "node",
13      "id": 260176446,
14      "lat": 48.6922167,
15      "lon": 8.9509296,
16      "tags": {
17        "amenity": "restaurant",
18        "name": "Heisser Ofen",
19        "wheelchair": "yes"
20      }
21    },
22    {
23      "type": "node",
24      "id": 286868069,
25      "lat": 48.6932258,
26      "lon": 8.9607236,
27      "tags": {
28        "amenity": "restaurant",
29        "cuisine": "pizza",
30        "name": "Ristorante Pizzeria Da Pippo",
31        "wheelchair": "yes"
32      }
33    },
34    {
35      "type": "node",
36      "id": 295728634,
37      "lat": 48.6692250,
38      "lon": 9.0052570,
39      "tags": {
40        "addr:city": "Böblingen",
41        "addr:country": "DE",
42        "addr:housenumber": "59",
43        "addr:postcode": "71034",
44        "addr:street": "Freiburger Allee",
45        "amenity": "restaurant",
46        "name": "Smart",
47        "toilets:wheelchair": "no",
48        "website": "http://www.restaurant-smart.com",
49        "wheelchair": "limited"
50      }
51    }
52  ]
53  }
```

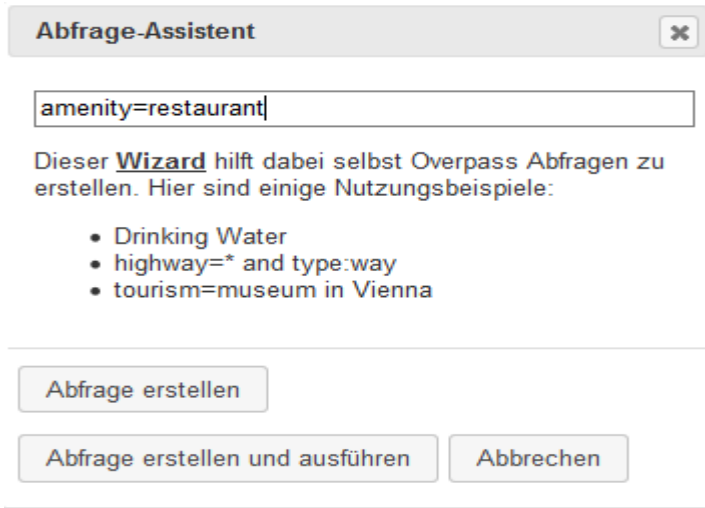

Weiterentwicklung der Abfrage

1. `[out:json][timeout:25];`
2. `// fetch area "Böblingen" to search in`
3. `{{geocodeArea:Böblingen}}->.searchArea;`
4. `// gather results`
5. `(`
6. `nwr["amenity"="restaurant"][website](area.searchArea);`
7. `);`
8. `// print results`
9. `out body;`
10. `>; out skel qt;`

Weiterentwicklung der Abfrage

1. **[out:json][timeout:25];**
2. // fetch area "Böblingen" to search in
3. **{{geocodeArea:Böblingen}}->.searchArea;**
4. // gather results
5. (
6. **nwr["amenity"="restaurant"][website](area.searchArea);**
7. **nwr[tourism=hotel][website](area.searchArea);**
8. **);**
9. // print results
10. **out body;**
11. **>; out skel qt;**

die Bounding Box



Abfrage-Assistent

amenity=restaurant

Dieser **Wizard** hilft dabei selbst Overpass Abfragen zu erstellen. Hier sind einige Nutzungsbeispiele:

- Drinking Water
- highway=* and type:way
- tourism=museum in Vienna

Abfrage erstellen

Abfrage erstellen und ausführen

Abbrechen

```
1. [out:json][timeout:25];
2. // gather results
3. (
4.   // query part for: "amenity=restaurant"
5.   node["amenity"]="restaurant"({{bbox}});
6.   way["amenity"]="restaurant"({{bbox}});
7.   relation["amenity"]="restaurant"({{bbox}});
8. );
9. // print results
10. out body;
11. >;
12. out skel qt;
```

Verknüpfungen

- Union

1. (
2. nwr[amenity=restaurant][website](area.searchArea);
3. nwr[tourism=hotel][website](area.searchArea);
4.);

- Differenz

1. (
2. nwr[amenity=restaurant](area.searchArea);
3. - nwr[amenity=restaurant][website](area.searchArea);
4.);

Ausgabe

- **Format**

[out:json]

[out:xml]

[out:csv(fieldname_1 [,fieldname_n ...] [; csv-header [; csv-separator]])]

[out:csv(::id,::type,name,cuisine,website)]

- **Daten**

out body;

ids: Print only the ids of the elements in the set

skel: Print the minimum information necessary for geometry

body: Print all information necessary to use the data

tags: Print only ids and tags for each element

meta: Print everything known about the elements

count: Print number of found elements

>; out skel qt;

Ausgabe

AusführenTeilenExportWizardSpeichernLadenAusloggenEinstellungenHilfe

overpass turbo

KarteDaten

1

[out:csv(::id,::type,name,cuisine,website)]

2

[timeout:25];

3

// fetch area "Böblingen" to search in

4

{{(geocodeArea:Böblingen)}}->.searchArea;

5

// gather results

6

(

7

nwr[amenity=restaurant][website]

8

(area.searchArea);

9

);

10

// print results

11

out body;

//>; out skel qt;

1

@id @type

2

295728634 node

3

309830817 node

4

309832135 node

5

309833945 node

6

310103886 node

7

311806875 node

8

319598937 node

9

320950594 node

10

326286773 node

11

393195495 node

12

453643812 node

13

520116030 node

14

635744397 node

15

1670712090 node

16

1891963208 node

17

2166857853 node

18

2626768420 node

19

2903496876 node

20

3363277091 node

21

3654716027 node

22

3866994696 node

23

4532944550 node

24

6053081941 node

25

6151274539 node

26

6423971742 node

27

6444510786 node

28

7002952885 node

29

7127850842 node

30

236484430 way

31

32641270 way

32

32642941 way

33

34353131 way

34

40523030 way

35

42276441 way

36

48001086 way

37

48003510 way

38

48304318 way

39

48424516 way

40

277524325 way

41

cuisine website

Smart http://www.restaurant-smart.com

Ehrbar burger http://ehrbar-bb.de/

Platzhirsch german;regional https://www.platzhirsch-boeblingen.de/

Bootshaus am Oberen See german http://www.bootshaus-boeblingen.de

El Zapata mexican http://www.el-zapata.com

Sinam chinese;vietnamese http://www.si-nam.de

Neuberths am See http://www.neuberths-am-see.de

Asia 5 Sterne asian https://boeblingen.asia5sterne.de

Da Signora italian http://www.ristorante-dasignora.de/

IBM Klub Restaurant balkan;german;pasta http://ibm-klubrestaurant.com

Paladion https://www.heilingsgastro.de/im-paladion

Hola Mexicana mexican https://www.hola-mexicana.de

Wichtel https://www.wichtel.de/

ESV Gaststätte http://www.amtisch.de/eisenbahner.htm

Thermini german https://www.heilingsgastro.de/de/thermini

Mauritius https://www.my-mauritius.com/restaurants/boeblingen

Tower 66 - Steakhouse & Bar steak_house https://www.tower66-steakhouse.de/

Il Fresco https://www.il-fresco.de/

Restaurant Merhaba mediterranean https://www.restaurant-merhaba.de/

Balaton zum Schlosskeller hungarian http://www.balatonzumschlosskeller.de

Ristorante Pizzeria da Michelina italian http://www.ristorante-da-michelina.de/

Check Inn Foodport international http://www.check-inn.events

Indian King indian http://desi-tadka.de

Café Lichtwerk https://cafe-lichtwerk.de/

Restaurant Gaumenfreude regional https://www.restaurant-gaumenfreude.de/

Tokyo Sushi sushi http://www.sushibb.de

Kitchen https://www.kitchen-bb.de

Suite 25 https://www.suite25.de/

"Restaurant "Gauri"" german;american http://restaurant-zimmerschlag.de

Da Gianni italian;pizza https://www.da-gianni.com

Ming chinese https://restaurant-ming.eatbu.com

Zum Reussenstein regional https://www.reussenstein.com

Ganssee Stüble italian;pizza;german http://www.gansseestueble.de

Dream Bowl american https://www.dreambowl.de/dream-bowl-boeblingen

Seegärtle burger;german;breakfast;american;ice_cream;coffee_shop;cake http://www.seegaertle.de

Frechdax https://www.frech-bb.de/125/ladengeschaefte/frechdax

Schönbuch Bräu Brauhaus https://www.brauhaus-bb.de

Schönaicher Fürst http://www.derfuerst.net/index.htm

Ristorante Pizzeria Ochsentrog pizza https://www.restaurant-ochsentrog.de/

Ausgabe

Ausführen Teilen Export Wizard Speichern Laden Ausloggen Einstellungen Hilfe overpass turbo

Karte Daten

```
1 [out:json]
2 [timeout:25];
3 // fetch area "Böblingen" to search
4 in
5 {{geocodeArea:Böblingen}}->.searchArea
6 a;
7 // gather results
8 way[amenity=restaurant][website]
9 {{.searchArea}};
10 // print results
11 out body;
12 //>> out skel qt;
```

Diese Karte ist leer. (no visible data)

Mercedes-Benz Werk Sindelfingen

Böblingen

geladen – Nodes: 0, Ways: 11, Relations: 0
angezeigt – POIs: 0, Linien: 0, Polygone: 0

verpass-turbo.eu/#close

Kommandos

Ausführen

Teilen

Export

Wizard

Speichern

Laden

Ausloggen

Einstellungen

Hilfe

Teilen

Permalink

Kopiere diesen Link, um die aktuelle Abfrage zu teilen:

<http://overpass-turbo.eu/s/TLF>

Optionen

- ☒ Aktuelle Kartenansicht mitgeben
- ☐ Die geteilte Abfrage sofort beim Öffnen ausführen

Erledigt

Exportieren

▼ Daten

Speichere/Kopiere als GeoJSON

Speichere/Kopiere als GPX

Speichere/Kopiere als KML

Speichere/Kopiere als OSM Rohdaten

Rohdaten direkt von Overpass API

In einen OSM-Editor laden: JOSM,

Level0

► Karte

► Abfrage

Erledigt

Speichern

Name für diese Abfrage:

Auf osm.org speichern

Speichern (lokal)

Abbrechen

Warnungen

- Syntax muss exakt eingehalten werden
- Fehlermeldungen teils kryptisch
- Einschließen von keys/values in "" schadet nie
- Verwendung von „;“ teils ungewöhnlich
- Vorsicht bei großen Datenmengen: Browser „stirbt“
- Bei Kartenausgabe `> out skel qt;` nicht vergessen
- Kommentare `//, /* */` machen das Leben leichter
- Overpass hat eigene Datenbasis: einige Min. Lag

weitere Themen

- Value matches regular expression (~, !~)
- Area (ähnlich geocoding)

```
( area[admin_level=6][name="Landkreis Ludwigsburg"];  
  area[admin_level=6][name="Landkreis Böblingen"];  
  area[admin_level=6][name="Landkreis Esslingen"];  
  area[admin_level=6][name="Rems-Murr-Kreis"];  
  area[admin_level=6 ][name="Stuttgart"]; ) ->.sarea;
```
- Verwendung von Variablen
- Einbindung in Tools: JOSM, UMAP (Heiko 2019)
- Zeitabfragen (newer, date, diff)
- Evaluators
- CSS

regular expression / Variablen

```
[timeout:180];
```

```
nwr[~"^.*:covid19$"~"."]({{bbox}});
```

```
out meta;
```

```
{{LK="Landkreis Esslingen"}}
```

```
area[admin_level=6][name={{LK}}]->.searchArea;
```

Nachtrag

```
nwr[~"^\.*:covid19$"~".""]({{bbox}});
```

Die Frage war, ob es `~"."` oder `~".*"` heißen muss.

Lt. https://wiki.openstreetmap.org/wiki/Overpass_API/Overpass_QL ist beides möglich.

Das macht auch Sinn, wenn man den Operator `~` als „enthält“ deutet.

Es werden sowohl keys mit mindestens einem beliebigen Zeichen (".") oder einer beliebigen Zeichenkette (".*") mit mindestens einem Zeichen gefunden.

Beides ist in diesem Fall identisch.

Nachtrag: > oder >>

Recurse down (>)

The *recurse down* standalone query is written as a single greater than. It takes an input set. It produces a result set. Its result set is composed of:

- all **nodes** that are part of a **way** which appears in the **input** set; plus
- all **nodes and ways** that are members of a **relation** which appears in the **input** set; plus
- all **nodes** that are part of a **way** which appears in the **result** set

Recurse down relations (>>)

The *recurse down relations* statement has a similar syntax to the *recurse down* statement, but it is written as a double greater than.

In particular, you can change the input and/or result set with the same notation as for the *recurse-down* standalone query.

It continues to follow the membership links including nodes in ways until for every object in its input or result set all the members of that object are in the result set as well.

Precisely, the *recurse down relations* statement returns the transitive and reflexive closure of membership.

Nachtrag: > oder >>

```
relation[name="Martinusweg Hauptweg"][type=superroute];  
out meta;  
>>; out skel qt;  
// die gefundenen Relation ist eine Superrelation mit Relationen als Members  
// daher ist >>; notwendig
```

```
relation[name="Martinusweg HW-35-11 Herrenberg -  
Böblingen"][type=route];  
out meta;  
>; out skel qt;  
// die gefundenen Relation ist eine einfache Relation mit nodes und ways als Members  
// daher ist >; ausreichend, >>; tuts aber auch
```