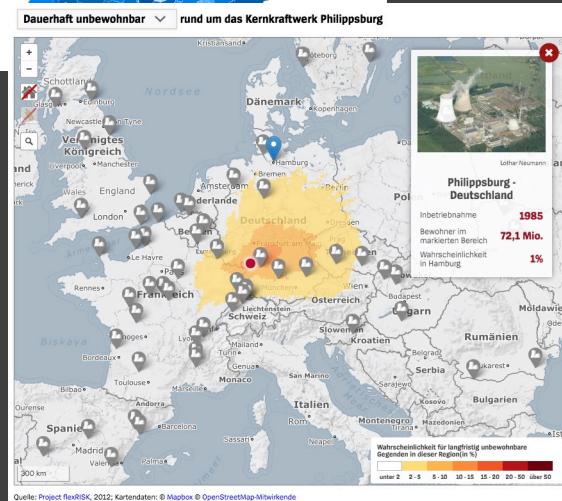
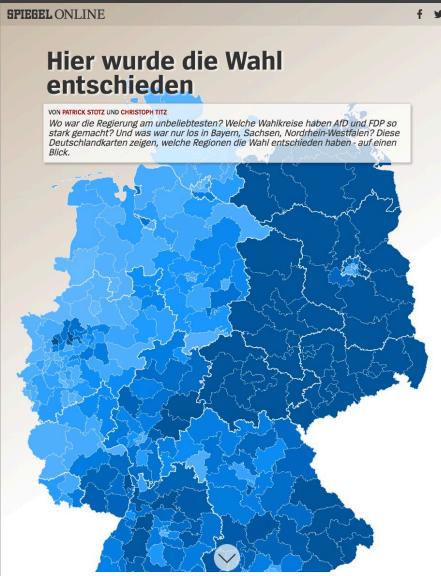


Geodaten im Web - eine Sammlung kleiner Tools rund um das GeoJSON-Format

@PatrickStotz

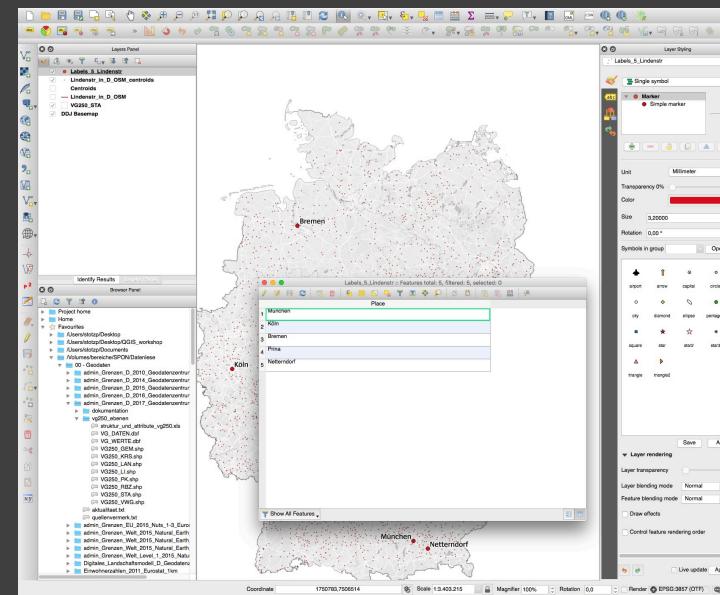
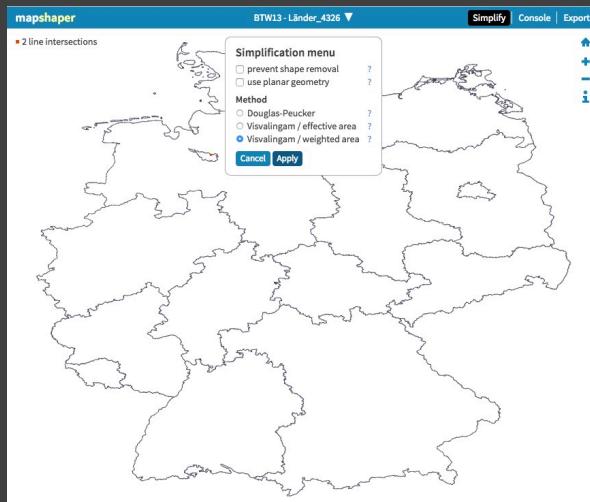
@SPIEGEL_data

Karten und Geodaten im Datenjournalismus



Warum kleine Tools?

- kein komplizierte GIS-Nutzeroberfläche
- erledigen genau einen Job
- benötigen keine Installationen
- gut geeignet für Nicht-Experten (Grafiker, Programmierer)



Warum GeoJSON?

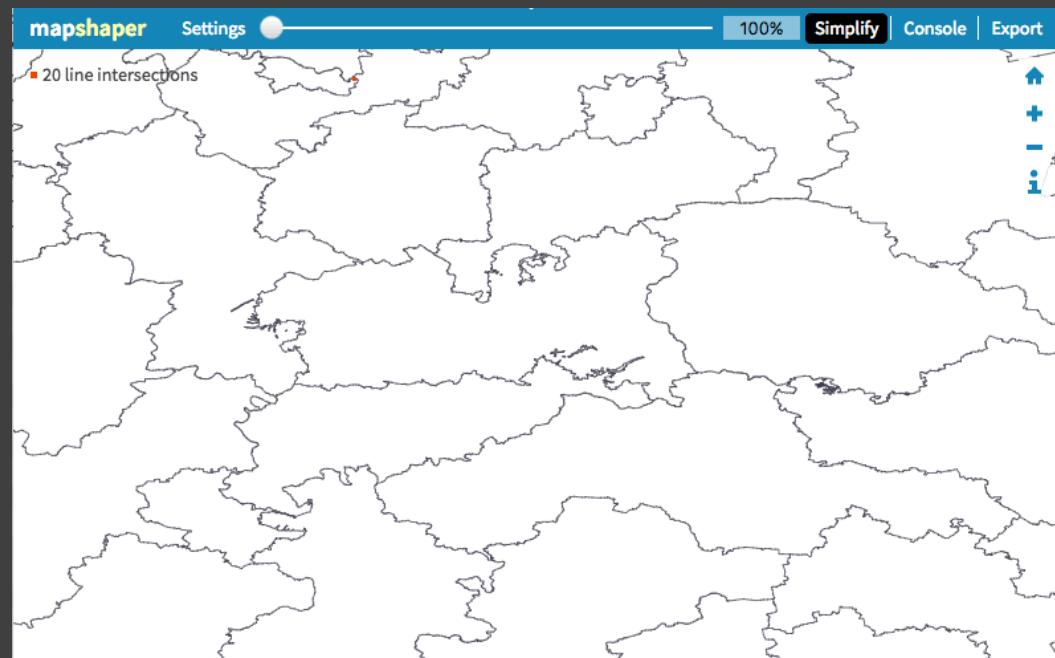
- unsere Endprodukte sind JavaScript-Anwendung
- human-readable
- Versionskontrolle
- alles in einer Datei
- Tom McWright: [More than you ever wanted to know about GeoJSON](#)
- offizielle Spezifikation

```
{ "type": "FeatureCollection",
  "features": [
    { "type": "Feature",
      "geometry": {
        "type": "Point",
        "coordinates": [102.0, 0.5]
      },
      "properties": {
        "prop0": "value0"
      }
    },
    { "type": "Feature",
      "geometry": {
        "type": "LineString",
        "coordinates": [
          [102.0, 0.0], [103.0, 1.0], [104.0, 0.0], [105.0, 1.0]
        ]
      },
      "properties": {
        "prop0": "value0",
        "prop1": 0.0
      }
    },
    { "type": "Feature",
      "geometry": {
        "type": "Polygon",
        "coordinates": [
          [ [100.0, 0.0], [101.0, 0.0], [101.0, 1.0],
            [100.0, 1.0], [100.0, 0.0] ]
        ]
      },
      "properties": {
        "prop0": "value0",
        "prop1": {"this": "that"}
      }
    }
  ]
}
```

mapshaper

Simplifizieren von Geodaten, kann auch Attribute editieren, clippen, dissolven, filtern...

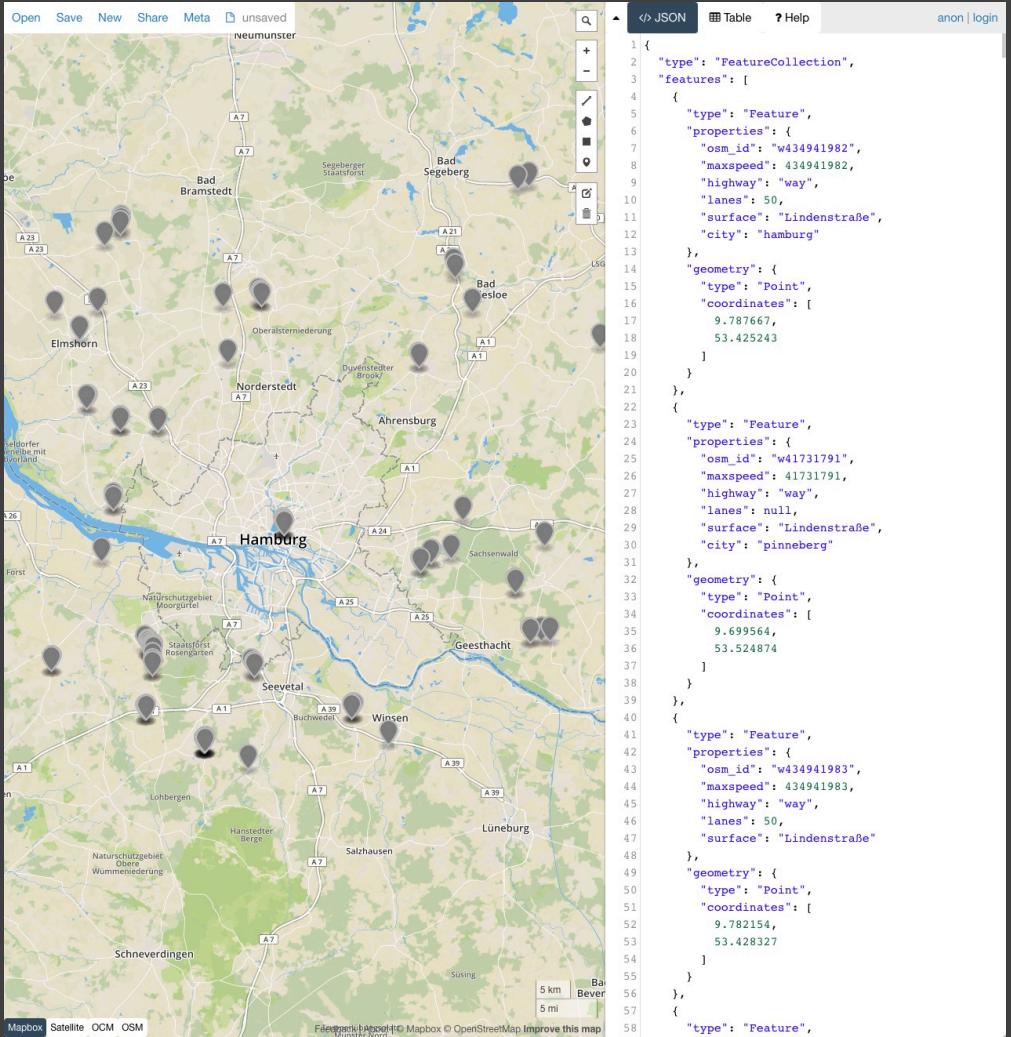
- ist gleichermaßen Nutzeroberfläche im Web, als auch Kommandozeilen-Tool
- direktes visuelles Feedback
- Berechnung findet im Browser statt (Daten bleiben privat)
- Export in u.a. SVG
- Projektdokumentation



geojson.io

Geodaten anzeigen und erkunden

- Features hinzufügen/löschen
- Features editieren (Attribute und Geometrien)



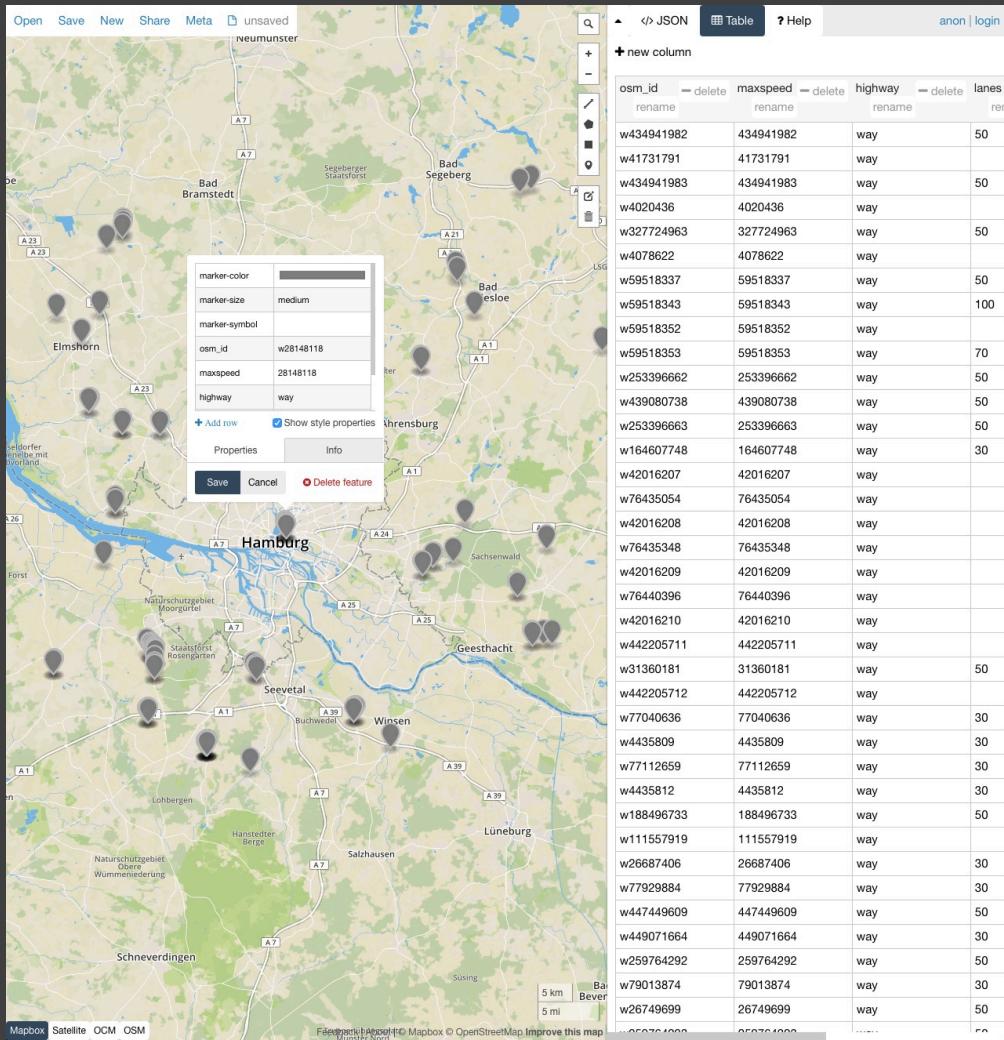
The screenshot shows a map of the Hamburg region in Germany, centered around the city of Hamburg. The map includes labels for various towns and regions such as Neumünster, Bad Bramstedt, Elmshorn, Norderstedt, Ahrensburg, Geesthacht, Winsen, Lüneburg, and Schnetverdingen. Several geographical features are highlighted with black callout bubbles, including a highway labeled "Lindenstraße" near Hamburg, a nature reserve labeled "Naturschutzgebiet Moorgürtel" near Rosengarten, and a forest area labeled "Staatsforst Rosengarten". The map interface includes a top bar with "Open", "Save", "New", "Share", "Meta", and "unsaved" buttons, and a sidebar with zoom controls (+, -, fit, location). On the right side, there is a JSON editor window displaying the geodata for the highlighted features. The JSON code is as follows:

```
1 {
2   "type": "FeatureCollection",
3   "features": [
4     {
5       "type": "Feature",
6       "properties": {
7         "osm_id": "w434941982",
8         "maxspeed": 434941982,
9         "highway": "way",
10        "lanes": 50,
11        "surface": "Lindenstraße",
12        "city": "hamburg"
13      },
14      "geometry": {
15        "type": "Point",
16        "coordinates": [
17          9.787667,
18          53.425243
19        ]
20      }
21    },
22    {
23      "type": "Feature",
24      "properties": {
25        "osm_id": "w41731791",
26        "maxspeed": 41731791,
27        "highway": "way",
28        "lanes": null,
29        "surface": "Lindenstraße",
30        "city": "pinneberg"
31      },
32      "geometry": {
33        "type": "Point",
34        "coordinates": [
35          9.699564,
36          53.524874
37        ]
38      }
39    },
40    {
41      "type": "Feature",
42      "properties": {
43        "osm_id": "w434941983",
44        "maxspeed": 434941983,
45        "highway": "way",
46        "lanes": 50,
47        "surface": "Lindenstraße"
48      },
49      "geometry": {
50        "type": "Point",
51        "coordinates": [
52          9.782154,
53          53.428327
54        ]
55      }
56    },
57    {
58      "type": "Feature",
59    }
60  ]
61}
```

geojson.io

Geodaten anzeigen und erkunden

- Features hinzufügen/löschen
- Features editieren (Attribute und Geometrien)



geojson.io

Geodaten teilen
per URL und mit Kartenposition

`geojson.io/#id=gist:anonymous/e5fc0bf7c420bd8dc17aba3a2c63c93d&map=10/53.6015/10.0443`

- anonym als privates GitHub gist
- oder nach Login im eigenen GitHub-Account

geojson.io

offen vorliegende Geodaten per URL-Aufruf anzeigen

<http://api.tiles.mapbox.com/v3/tmcw.map-gdv4cswo/markers.geojson>

<http://geojson.io/#data=data:text/x-url,http%3A%2F%2Fapi.tiles.mapbox.com%2Fv3%2Ftmcw.map-gdv4cswo%2Fmarkers.geojson>

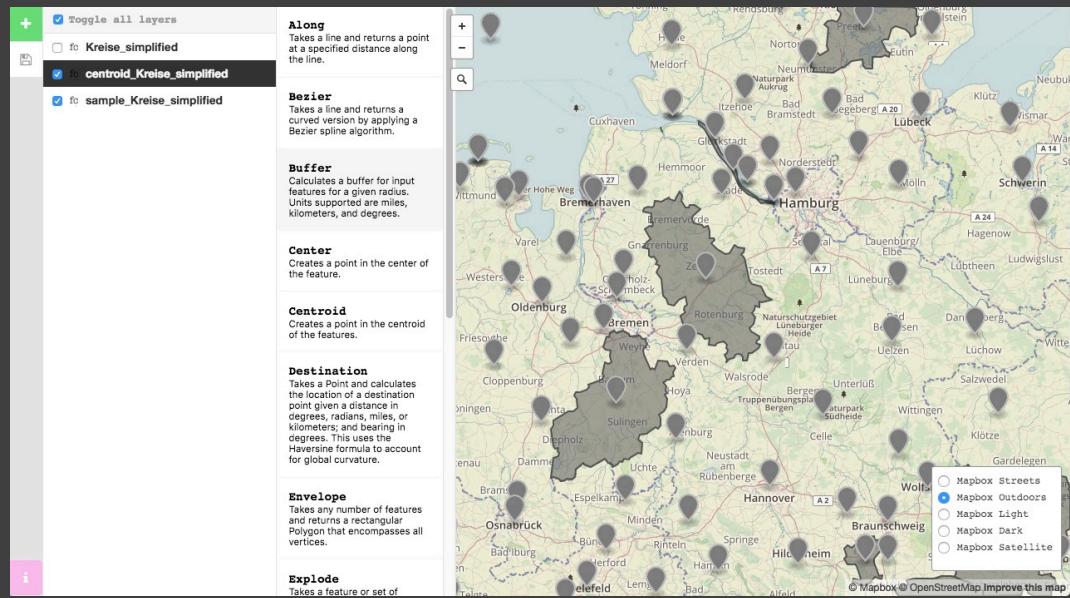
The screenshot shows a map of North America with numerous red location markers. The map includes state/province boundaries and major city labels. To the right of the map is an open JSON editor window titled 'anon / login'. The JSON code represents a FeatureCollection with several features, each containing a point geometry and a property object with a title, description, and marker details. The JSON code is as follows:

```
<div>Open Save New Share Mark unshare</div><div>Table Help</div><div><pre>1  {
2   "type": "FeatureCollection",
3   "features": [
4     {
5       "type": "Feature",
6       "properties": {
7         "id": "c19bshun7khaejriven7l8ep",
8         "title": "Cornell University",
9         "description": "<div style='width:330px;height:240px;overflow:auto;'><div class='googlt-info-window'>Cornell University</div></div>",
10        "marker-color": "#ff0000",
11        "marker-size": "medium",
12        "marker-symbol": "airport",
13        "marker-zoom": ""
14      },
15      "geometry": {
16        "coordinates": [
17          -76.922448,
18          42.753559
19        ]
20      }
21    },
22    {
23      "type": "Point"
24    },
25    {
26      "type": "Feature",
27      "properties": {
28        "id": "c19bshun7khaejriven7l8ep",
29        "title": "Vivisoxjet",
30        "description": "<div style='width:330px;height:240px;overflow:auto;'><div class='googlt-info-window'>Vivisoxjet</div></div>",
31        "marker-color": "#ff0000",
32        "marker-size": "medium",
33        "marker-symbol": "airport",
34        "marker-zoom": ""
35      },
36      "geometry": {
37        "coordinates": [
38          -75.7535,
39          34.303928
40        ]
39      }
40    },
41    {
42      "type": "Point"
43    },
44    {
45      "type": "Feature",
46      "properties": {
47        "id": "c19bshun7khaejriven7l8ep",
48        "title": "Air Force - 2011-NDA-55-COA",
49        "description": "<div style='width:330px;height:240px;overflow:auto;'><div class='googlt-info-window'>Air Force - 2011-NDA-55-COA</div></div>",
50        "marker-color": "#ff0000",
51        "marker-size": "medium",
52        "marker-symbol": "airport",
53        "marker-zoom": ""
54      },
55      "geometry": {
56        "coordinates": [
57          -123.915132,
58          44.216494
59        ]
58      }
59    }
60  ]
61}</pre></div>
```

dropchop.io

Einfache Geo-Operationen im Browser

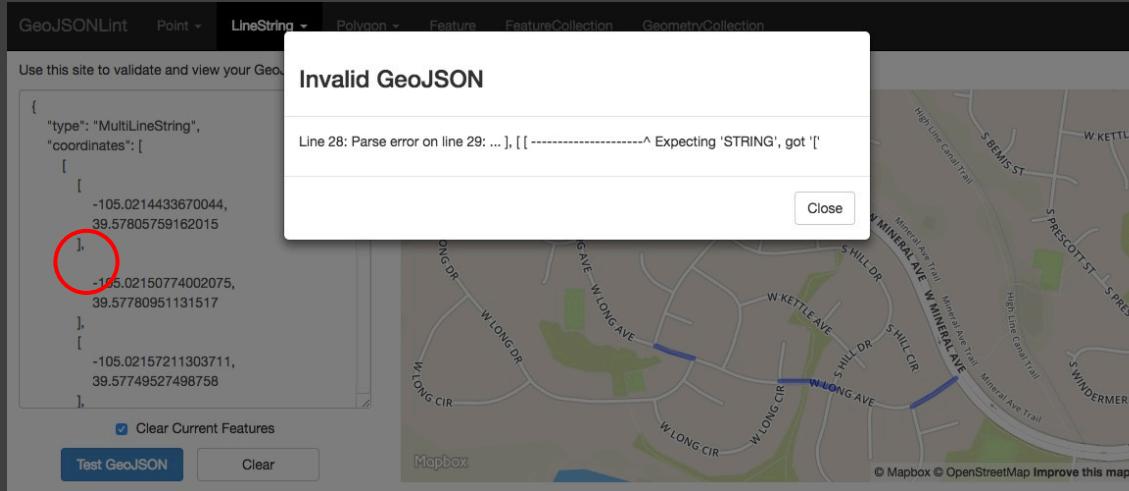
- Zentroide ermitteln
- Buffer zeichnen
- Stichprobe ziehen
- Simplifizieren
- Abfrage Punkte in Fläche
- Basiert auf turf.js



GeoJSONLint

Geodaten validieren

- hilft Fehler in kaputten GeoJSON-Dateien zu finden
- identifiziert fehlerhafte Features
- Fehlermeldungen sind etwas kryptisch



ogre – ogr2ogr web client

Dateiformate umwandeln

- basiert auf ogr2ogr
- verschiedene Geodaten-Formate in GeoJSON umwandeln
- auch GeoJSON → Shapefile

Type	Single File Format	Zipped (.zip) File Format
BNA	.bna	-
CSV	.csv	-
DGN	.dgn	-
DXF	.dxf	-
ESRI Shapefile	-	.shp, .dbf, and .shx (.prj optional)
GeoConcept	.gxt or .txt	-
GeoJSON	.json or .geojson	-
GeoRSS	.rss, .georss, or .xml	-
GML	.gml	.gml and .xsd
GMT	.gmt	-
GPX	.gpx	-
Interlis 1	.itf	.itf and .ili (.fmt optional)
KML / KMZ	.kml or .kmz	-
MapInfo	-	.tab, .map, .id, and .dat
S-57	.000	.000 (.001-00N optional)
TIGER	-	at least an .rt1
VRT	-	.vrt with any other spatial file

Mehr Tools: awesome-geojson

awesome geojson

GeoJSON utilities that will make your life easier.

operations

- **CoGran**: Resample statistics from one set of GeoJSON polygons to another
- **geo_assigner**: Copy a property to all intersecting features
- **geojson-coords**: Extract coordinates from GeoJSON
- **geojson-extent**: compute the bounding box of geojson features
- **geojson-flatten**: flatten multi geometries into normal geometries
- **geojson-multiply**: merge normal geojson features into one multi geometry type feature
- **geojson-js-utils**: JavaScript helper functions for manipulating GeoJSON
- **geojson-merge**: Merge multiple GeoJSON files into one FeatureCollection.
- **geojson-normalize**: normalize any geojson object into a geojson featurecollection
- **geojson-pick**: remove all but specified properties from features in a geojson featurecollection
- **geojson-random**: generate random geojson points, lines, and polygons
- **geojson-rewind**: enforce ring winding order
- **geojson-summary**: get a plain-english summary of what's in a geojson file
- **point-grouper**: group geojson points into containing polygons
- **geojson-join**: join geojson against json, dbf, and csv files
- **simplify-geojson**: apply the ramer-douglas-peucker line simplification to geojson features or feature collections in JS or on the CLI
- **turf**: collection of functions for spatial operations and analysis
- **winnow**: run sql queries against geojson with javascript

editors & viewers

- **geojson.io**: web-based editor, supports many filetype imports & exports, operations, sharing via GitHub
- **umap**: web-based editor, supports sharing on-site
- **simple geojson editor**: geojson editor on a google map, by google
- **mapstarter**: helps generate svg, images, and code from GeoJSON
- **giv**: electron app enables viewing GeoJSON locally and hack on it with turf; works offline
- **GeoJSON editor**: a modified version of Google's simple geojson editor
- **geojson2image**: library for generating images from GeoJSON
- **dropchop**: browser-based GIS based on Turf.js

awesome geojson

GeoJSON utilities that will make your life easier.

operations

- **CoGran**: Resample statistics from one set of GeoJSON polygons to another
- **geo_assigner**: Copy a property to all intersecting features
- **geojson-coords**: Extract coordinates from GeoJSON
- **geojson-extent**: compute the bounding box of geojson features
- **geojson-flatten**: flatten multi geometries into normal geometries
- **geojson-multiply**: merge normal geojson features into one multi geometry type feature
- **geojson-in-utility**: JavaScript helper functions for manipulating GeoJSON
- **geojson-normalize**: normalize any geojson object into a geojson featurecollection
- **geojson-pick**: remove all but specified properties from features in a geojson featurecollection
- **geojson-random**: generate random geojson points, lines, and polygons
- **geojson-rewind**: enforce ring winding order
- **geojson-summary**: get a plain-english summary of what's in a geojson file
- **geojson-join**: join geojson against json, dbf, and csv files
- **simplify-geojson**: apply the ramer-douglas-peucker line simplification to geojson features or feature collections in JS or on the CLI
- **turf**: collection of functions for spatial operations and analysis
- **winnow**: run sql queries against geojson with javascript

editors & viewers

- **geojson.io**: web-based editor, supports many filetype imports & exports, operations, sharing via GitHub
- **umap**: web-based editor, supports sharing on-site

simple geojson editor: geojson editor on a google map, by google

mapstarter: helps generate svg, images, and code from GeoJSON

giv: electron app enables viewing GeoJSON locally and hack on it with turf; works offline

GeoJSON editor: a modified version of Google's simple geojson editor

geojson2image: library for generating images from GeoJSON

dropchop: browser-based GIS based on Turf.js

validation

- **geojsonlint**: find errors in your geojson files

services

- **geojsonio-cgi**: send geojson features to geojson.io from your command line

geojsonio-extension: chrome extension for editing gitlab files in geojson.io

geojsonlint: REST interface for GeoJSON validation

geojson-server: simple interface for simplification and conversion of GeoJSON and TopoJSON

geojson-server: keep Server with plugins that read Esri, Github, Socrata and other services in GeoJSON endpoints

featureserver: An open source Esri-Style Feature Server

conversion

- **co2geojson**: convert CSV to geojson

geojson-to-mapnik: Transform GeoJSON objects into Mapnik XML, styleSheets with embedded GeoJSON data and simpleKey-style-derived styles

geojson-vt: Slice GeoJSON into vector tiles on the fly in the browser

geojson2vt: convert geojson to vt and vtv

geojson2vtv: convert geojson to vtv (using terrainify) from a set of geojson features

geojson2vtv: convert anything to anything

- **gis2**: nice python interface on top of ogr

minar: converts OpenStreetMap data to GeoJSON faster than anything else

osm2geojson: convert OpenStreetMap data to streaming GeoJSON

topojson: convert topojson to geojson

turf: convert geojson to KML

topojson: convert GeoJSON to a kml TopoJSON, join data from CSV

osm2geojson: convert OpenStreetMap vector tiles

wellknown: convert vt to geojson

osm2geojson: convert OpenStreetMap data to GeoJSON

osm2geojson: convert OSM XML data to GeoJSON

geojson2raster: convert any size of geojson into a streaming csv, shapefile or kmz, etc.

supercluster: A fast GeoJSON clustering library for browsers and Node

rte7945-to-d3: Converts polygon ordering under rte7945 to d3 compatible conventions

geojson-to-svg: Convert GeoJSON geometry paths to SVG polyline elements

geojson-to-svg: Command line tool to convert GeoJSON to SVG

data

natural-earth: country, province, and geographical data

geojson-vt: natural earth in web-friendly sizes with tiling

world-atlas: customizable simplified versions of natural earth data

geo-maps: High Quality GeoJSON maps programmatically generated

topojson2geojson: convert topojson to geojson

us-states: geojson & topojson for United States features

metaclicks: regional OpenStreetMap data as GeoJSON

airportsnear-airport: outline of every airport

airportsnear-airport: outline of every airport

whereworth-names: other features output from Geofishnet by Aaron Stroup

topo-geonames: The Getty Thesaurus of Geographic Names (TGN) As GeoJSON

strava-geojson: export runs & rides from Strava to GeoJSON

strava-geojson: export all strava data to geojson, in node & on the web

serialization

geojson-serializers: serializes geojson to their python datatypes

open-geojson: serializes geojson to their ruby datatypes

rest-geojson: serializes geojson to their rest datatypes

geojson-jackson: serializes GeoJSON to and from Java datatypes (based on Jackson)

moshion-jackson: serializes GeoJSON to and from Java datatypes (based on GSON)

resources

IFC 744M - The GeoJSON Format: the current GeoJSON standard by IETF

GeoJSON.org: the first specification, defining all the rules for GeoJSON structures

GeoJSON: more than you ever wanted to know about GeoJSON: a review of the concepts behind the specification in a more narrative form

Zeit für Fragen

Haben Sie Hinweise zu spannenden Datensätzen oder Forschungsergebnissen?

Kontakt

Patrick.Stotz@spiegel.de

[@PatrickStotz](https://twitter.com/PatrickStotz)