

Web-APIs für geographische Anwendungen

Jakob Miksch

meggsimum - Büro für Geoinformatik

FOSSGIS 2022, 09.03.2022

Jakob Miksch



- Geoinformatiker
- Anwendungsentwickler
- OSGeo Foundation Charter Member

✉ jakob@meggsimum.de

👤 [@jakobmiks](#)

🐦 [@jakobmiks](#)

meggsimum



✉ info@meggsimum.de
🐙 [@meggsimum](#)
🐦 [@meggsimum](#)


- meggsimum.de
- Dienstleistungen im Bereich GIS, Webmapping & GDI
- Maßgeschneiderte WebGIS-Lösungen
- Softwarekonzepte und Softwareentwicklung
- Geodaten
- Beratung und Schulung

(Web) APIs

- API = Application Programming Interface
- allgegenwärtig im Internet
- Datenaustausch
- Softwareentwicklung

Beispiel

OpenWeatherMap

 OpenWeather

Weather in your city

GuideAPIPricingMapsOur InitiativesPartnersBlogMarketplaceSign inSupport

Current Weather Data

API docSubscribe

- Access current weather data for any location including over 200,000 cities
- We collect and process weather data from different sources such as global and local weather models, satellites, radars and a vast network of weather stations
- JSON, XML, and HTML formats
- Included in both free and paid subscriptions

Hourly Forecast 4 days

API docSubscribe

- Hourly forecast is available for 4 days
- Forecast weather data for 96 timestamps
- JSON and XML formats
- Included in the Developer, Professional and Enterprise subscription plans

One Call API

API docSubscribe

- Make one API call and get current, forecast and historical weather data
- **Minute forecast** for 1 hour
- **Hourly forecast** for 48 hours
- **Daily forecast** for 7 days
- **Historical data** for 5 previous days
- **National weather alerts**
- JSON format
- Included in both free and paid subscriptions

Daily Forecast 16 days

API docSubscribe

- 16 days forecast is available for any location or city
- 1-day step for 16 days
- JSON and XML formats
- Included in all paid subscription plans

Climatic Forecast 30 days

API docSubscribe

- Forecast weather data for 30 days
- JSON format
- Included in the Developer, Professional and Enterprise subscription plans

Bulk Downloading


API docSubscribe

- You may request current weather and forecasts in bulk with a variable data granulation
- Current weather bulk is available for 209,000+ cities
- Variety of hourly and daily forecast bulks depends on the frequency of data updating
- Additionally, this product allows to get archived current and forecasts weather data for 7 previous days
- Included in the Professional and Enterprise subscription plans

<https://openweathermap.org/api>

Beispiel

Google Street View

 Google Maps Platform

OverviewProductsPricingDocumentation▼BlogCommunity▼

SearchEnglish▼Sign in

Overview

Set up in Cloud Console

Using API Keys and Signatures

Image Metadata

Best Practices

Policies and Terms

Usage and Billing

Terms of Service

Street View Static API Policies

Other APIs

Maps JavaScript API

Maps Static API


Places Library, Maps JavaScript API

Maps Embed API

Maps URLs

Introduction

The Street View Static API lets you embed a static (non-interactive) Street View panorama or thumbnail into your web page, without the use of JavaScript. The viewport is defined with URL parameters sent through a standard HTTP request, and is returned as a static image.



`https://maps.googleapis.com/maps/api/streetview?size=400x400&location=47.5763831,-122.4211769&fov=80&heading=70&pitch=0&key=YOUR_API_KEY&signature=YOUR_SIGNATURE`

On this page

Introduction

Before you begin

URL parameters

Required parameters

Optional parameters

No imagery available



Show only valid images

More information

<https://developers.google.com/maps/documentation/streetview/overview>

Beispiel

NASA

 { APIs }		
Overview Generate API Key Authentication Recover API Key Browse APIs <input type="text"/> 		
Earth: Unlock the significant public investment in earth observation data		+
EONET: The Earth Observatory Natural Event Tracker		+
EPIC: Earth Polychromatic Imaging Camera		+
Exoplanet: Programmatic access to NASA's Exoplanet Archive database		+
GeneLab: Programmatic interface for GeneLab's public data repository website		+
Insight: Mars Weather Service API		+
Mars Rover Photos: Image data gathered by NASA's Curiosity, Opportunity, and Spirit rovers on Mars		+
NASA Image and Video Library: API to access the NASA Image and Video Library site at images.nasa.gov		+
TechTransfer: Patents, Software, and Tech Transfer Reports		+
Satellite Situation Center: System to cast geocentric spacecraft location information into a framework of (empirical) geophysical regions		+
SSD/CNEOS: Solar System Dynamics and Center for Near-Earth Object Studies		+
Techport: API to make NASA technology project data available in a machine-readable format		+
TLE API: Two line element data for earth-orbiting objects at a given point in time		+

<https://api.nasa.gov/index.html>

Austausch Formate

- XML
- JSON
- ...

XML

```
<breakfast_menu>
  <food>

  <name>Belgian Waffles</name>

  <price>$5.95</price>

  <description>
    Two of our famous Belgian Waffles with plenty of real
    maple syrup
  </description>

  <calories>650</calories>
  </food>
```

<https://www.w3schools.com/xml/simple.xml>

JSON

```
{  
  "squadName": "Super hero squad",  
  "homeTown": "Metro City",  
  "formed": 2016,  
  "secretBase": "Super tower",  
  "active": true,  
  "members": [  
    {  
  
  "name": "Molecule Man",  
  
  "age": 29,  
  
  "secretIdentity": "Dan Jukes",
```

<https://developer.mozilla.org/en-US/docs/Learn/JavaScript/Objects/JSON>

GeoJSON

```
{  
  "type": "Feature",  
  "geometry": {  
    "type": "Point",  
    "coordinates": [125.6, 10.1]  
  },  
  "properties": {  
    "name": "Dinagat Islands"  
  }  
}
```

<https://geojson.org/>

HTTP Methods

GET

POST

PUT

PATCH

DELETE

...

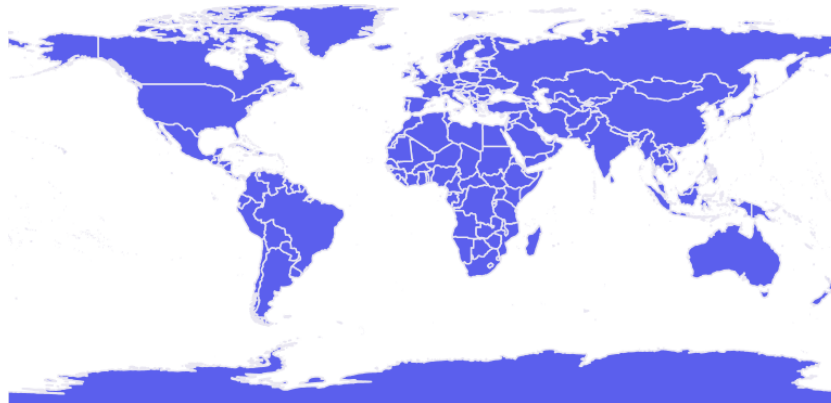
<https://developer.mozilla.org/en-US/docs/Web/HTTP/Methods>

URL - Query Parameter

```
https://www.example.com?  
color=blue&  
type=dark
```


Beispiel WMS

```
https://services.meggsimum.de/geoserver/mgsm-world/wms?  
service=WMS&  
version=1.1.0&  
request=GetMap&  
layers=mgsm-world:ne_10m_admin_0_countries&  
bbox=-180.0,-90.0,180.0,83.6341&  
width=768&  
height=370&  
srs=EPSG:4326&  
format=image/png
```



API Typen

- SOAP
- REST
- GraphQL
- ...

OGC Standards

- WMS, WFS, WCS, ...
- Beispiel: GeoServer (meggsimum)
- Weiterentwicklung: RESTful OGC API

Authentifizierung

typischerweise mit API-key

```
https://api.openrouteservice.org/v2/directions/driving-car?  
api_key=<--ADD_YOUR_API_KEY-->&  
start=8.681495,49.41461&  
end=8.687872,49.420318
```

Werkzeuge zum Zugriff

- Webbrowser
- Kommandozeile
- Programmiersprache
- Desktop Programme

Webbrowser

Entwicklerkonsole oder mit JavaScript

The screenshot displays the OpenStreetMap web application with a routing interface on the left and a browser developer console on the right.

OpenStreetMap Interface:

- Search Bar:** Coordinates 47.8180, 13.0189 and 47.8000, 13.0748 are entered. The mode is set to "Car (OSRM)".
- Directions Panel:** Shows a route from Münchner Bundesstraße (B155) to Salzburg. The distance is 5.5km and the time is 0:10. The route consists of 11 steps, including starting on Münchner Bundesstraße (B155), continuing on Ignaz-Harrer-Straße (B155), and ending on Clemens-Krauss-Straße.

Developer Console:

- Network Tab:** Shows a GET request to `routing.openstreetmap.de` with a status of 200. The response is a JSON object of size 10.99 KB.
- Response Tab:** Displays the JSON response structure, including waypoints and route details.

```
code: "Ok"
waypoints: [ [ {
  "hint": "TYcbga1stosAAAAHgAAABkAAAAKAAAAA_EVwP1HyAklw-9x8G6QAAAAAAeAAAAAGAAAAoAAAAV5gAA-6bGAAqL2QIUUp8YAEKXZAgEADwqXUHVn",
  "distance": 1.987312,
  "name": "Münchner Bundesstraße",
  "hint": "TYcbga1stosAAAAHgAAABkAAAAKAAAAA_EVwP1HyAklw-9x8G6QAAAAAAeAAAAAGAAAAoAAAAV5gAA-6bGAAqL2QIUUp8YAEKXZAgEADwqXUHVn",
  "distance": 1.987312
}, {
  "location": [ 13.018875, 47.817994 ],
  "name": "Münchner Bundesstraße"
}, {
  "hint": "qSO2i-AjtosyAAAAABgAAACsAAAAAAMjOHQn1cZkBNw-VBVSvrQTIAAAAGAAAAKwAAACwAAAAV5gAAcIHHALpe2QJwgccAwF7ZAgIAjwWXUHVn",
  "distance": 0.667114,
  "name": "Clemens-Krauss-Straße",
  "hint": "qSO2i-AjtosyAAAAABgAAACsAAAAAAMjOHQn1cZkBNw-VBVSvrQTIAAAAGAAAAKwAAACwAAAAV5gAAcIHHALpe2QJwgccAwF7ZAgIAjwWXUHVn",
  "distance": 0.667114
} ] ]
```

Kommandozeile

```
curl -X GET \  
'https://photon.komoot.io/api/?q=salzburg&limit=2'
```

ergibt:

```
{  
  "features": [  
    {  
      "geometry": {  
        "coordinates": [  
          13.0464806,  
          47.7981346  
        ],  
        "type": "Point"  
      },  
      "type": "Feature",  
      "properties": {  
        "osm_id": 86538.
```

Python

```
import requests

url = 'https://photon.komoot.io/api/?q=salzburg&limit=2'
response = requests.request('GET', url )

print(response)
```


R

```
library(httr)
library(jsonlite)

url = "https://photon.komoot.io/api/?q=salzburg"
res = GET(url)
data = fromJSON(rawToChar(res$content))

print(data)
```

JavaScript

```
const requestOptions = {  
  method: 'GET'  
};  
  
const url = "https://photon.komoot.io/api/?q=salzburg";  
fetch(url, requestOptions)  
  .then(response => response.text())  
  .then(result => console.log(result))  
  .catch(error => console.log('error', error));
```

QGIS

mit Python oder dem Debugging Panel

Unbenanntes Projekt - QGIS

Projekt Bearbeiten Ansicht Layer Einstellungen Erweiterungen Vektor Raster Datenbank Web Netz Verarbeitung Hilfe

Browser

- Favoriten
- Räumliche Lesezeichen
- Home
- C:\
- GeoPackage
- Spatialite
- PostGIS
- MSQL
- DB2
- WMS/WMTS
- Vector Tiles
- XYZ Tiles
- OpenStreetMap
- WCS
- WFS / OGC API - Features
- OWS
- ArcGIS-Map-Dienst
- ArcGIS-Feature-Dienst
- GeoNode

Layer

- OpenStreetMap

Diagnose-/Entwicklungswerkzeuge

Abfragen filtern

Anfragen

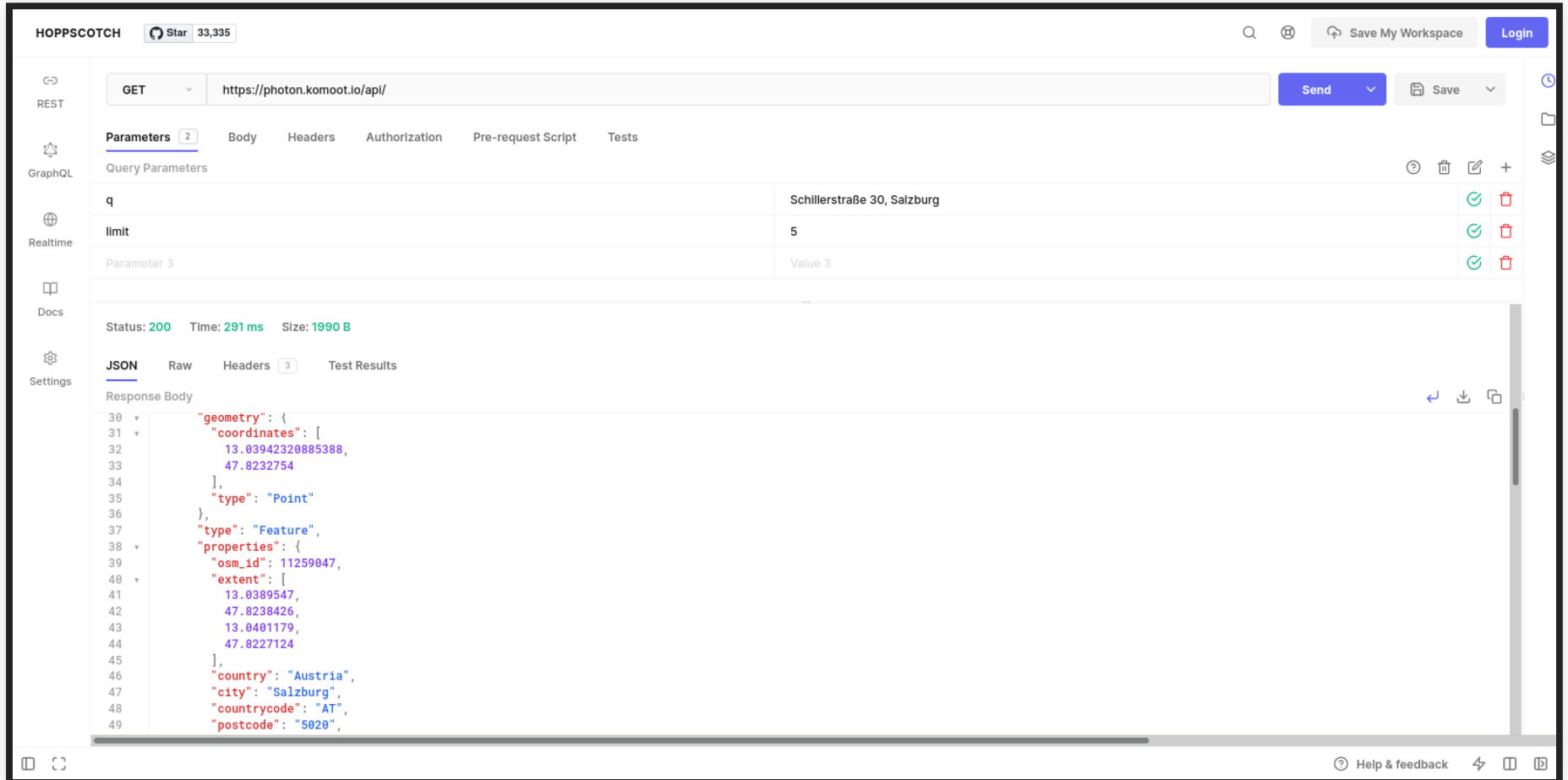
- 230 GET https://tile.openstreetmap.org/16/35142/22831.png
 - Anfrage
 - Operation : GET
 - Thread : 0x000001c0a3939070
 - Initiator : QgsWmsTiledImageDownloadHandler
 - ID : src\providers\wms\qgs_wms_provider.cpp:4195 (Qg-)
 - Cache (Steuerung) : Wenn verfügbar aus Cache laden, sonst aus dem-
 - Cache (Speicher) : Kann Ergebnis nicht in Cache speichern
 - Köpfe
 - Accept : */*
 - User-Agent : Mozilla/5.0 QGIS/31614
 - Antwort
 - Status : 200
 - Cache (Ergebnis) : Aus Netzwerk laden
 - Köpfe
 - Connection : keep-alive
 - Content-Length : 26634
 - Server : Apache/2.4.41 (Ubuntu)
 - Strict-Transport-Security : max-age=31536000; includeSubDomains; prelo-
 - Expect-CT : max-age=0
 - ETag : "f06e63caed19ead2c03efad5e671e6"
 - Cache-Control : max-age=15417, stale-while-revalidate=6048-
 - Expires : Sun, 21 Nov 2021 13:32:32 GMT
 - Access-Control-Allow-Origin : *
 - X-TileRender : odin.openstreetmap.org
 - Content-Type : image/png
 - Accept-Ranges : bytes
 - Date : Sun, 21 Nov 2021 12:38:16 GMT
 - Via : 1.1 varnish
 - Age : 12161
 - X-Served-By : cache-muc13934-MUC
 - X-Cache : HIT
 - X-Cache-Hits : 1
 - X-Timer : S1637498297.737837,V50,V1
- 231 GET https://tile.openstreetmap.org/16/35143/22830.png
- 232 GET https://tile.openstreetmap.org/16/35143/22831.png
- 233 GET https://tile.openstreetmap.org/16/35141/22830.png
- 234 GET https://tile.openstreetmap.org/16/35141/22831.png
- 235 GET https://tile.openstreetmap.org/16/35141/22829.png
- 236 GET https://tile.openstreetmap.org/16/35142/22829.png
- 237 GET https://tile.openstreetmap.org/16/35143/22829.png
- 238 GET https://tile.openstreetmap.org/16/35141/22832.png
- 239 GET https://tile.openstreetmap.org/16/35142/22832.png
- 240 GET https://tile.openstreetmap.org/16/35143/22832.png
- 241 GET https://tile.openstreetmap.org/16/35144/22829.png
- 242 GET https://tile.openstreetmap.org/16/35144/22830.png
- 243 GET https://tile.openstreetmap.org/16/35144/22831.png
- 244 GET https://tile.openstreetmap.org/16/35144/22832.png
- 245 GET https://tile.openstreetmap.org/16/35140/22829.png
- 246 GET https://tile.openstreetmap.org/16/35140/22830.png

Suche nach Typ (Strg+K)

Koordinate 1452641,6076766 Maßstab 1:4941 Vergrößerung 100% Drehung 0,0° Zeichnen EPSG:3857

Desktop Programme

Hoppscotch oder Postman



GeoCoding APIs

- OpenCage
- openrouteservice
- Photon
- Nominatim
- ...

Routing APIs

- GraphHopper
- openrouteservice
- OSRM (FOSSGIS Instanz)
- Valhalla (FOSSGIS Instanz)
- ...

Overpass API

Datenabfrage von OpenStreetMap (Beispiel)

Run

Share

Export

Wizard

Save

Load

Settings

Help

overpass turbo

Map

Data

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

Loaded — nodes: 210, ways: 13, relations: 1
Displayed — pois: 13, lines: 0, polygons: 12

Actinia API

Prozessierung von Geodaten mit GRASS GIS

OPERATIONS

Authentication Management

API Log

Cache Management

Satellite Image Algorithms

Vegetation index comput...

Query the Google Lands...

Download and import La...

Download and import Se...

NDVI computation of an ...

NDVI computation of an ...

Query the Google Sentin...

Generate the download u...

Location Management

Mapset Management

Processing

Raster Management

Raster Statistics

STRDS Management

STRDS Sampling

STRDS Statistics

Vector Management

Resource Management

SCHEMA DEFINITIONS

LocationListResponseModel

SimpleResponseModel

MapsetInfoResponseModel

ProcessLogModel

GrassModule

Satellite Image Algorithms

NDVI computation of an arbitrary Sentinel 2A scene.
The results are stored in the Google Cloud Storage.

PATH

POST /sentinel2_process_gcs/ndvi/{product_id}

DESCRIPTION

NDVI computation of an arbitrary Sentinel 2A scene. The processing is as follows: A user specific Sentinel 2A scene (Bands 04 and 08) will be download and imported into an ephemeral database.. The NDVI will be computed via r.mapcalc. The result of the computation is available as gzipped geotiff file. In addition, the univariate statistic will be computed as well as a preview image including a legend and scale. Minimum required user role: user.

REQUEST PARAMETERS

product_id	The product id of a sentinel scene
<div>required</div>	
In path	
string default:	
S2A_MSIL1C_20170212T104141_N0204_R008_T31TGJ_20170212T104138	

RESPONSES

200 OK	This response includes all created resources as
SentinelNDVIResponseModel	URL as well as the processing log and other metadata.
400 Bad Request	The error message and a detailed log why

Response Example (200 OK)

```
{
  "accept_datetime": "2018-05-30 12:25:43.987713",
  "accept_timestamp": 1527683143.9877105,
  "api_info": {
    "endpoint": "asyncephemeralsentinel2processingresource",
    "method": "POST",
    "path": "/api/v1/sentinel2_process"
```

Sonnenstand

```
http://voibos.rechenraum.com/voibos/voibos?  
name=sonnengang&  
Koordinate=-41390,228170&  
CRS=31254&  
Datum=11-21:12:34&  
H=2&  
Output=Horizont,Sonnenzeit
```

Dokumentation

Sonnenstand

Sonnengang mit Horizontdarstellung

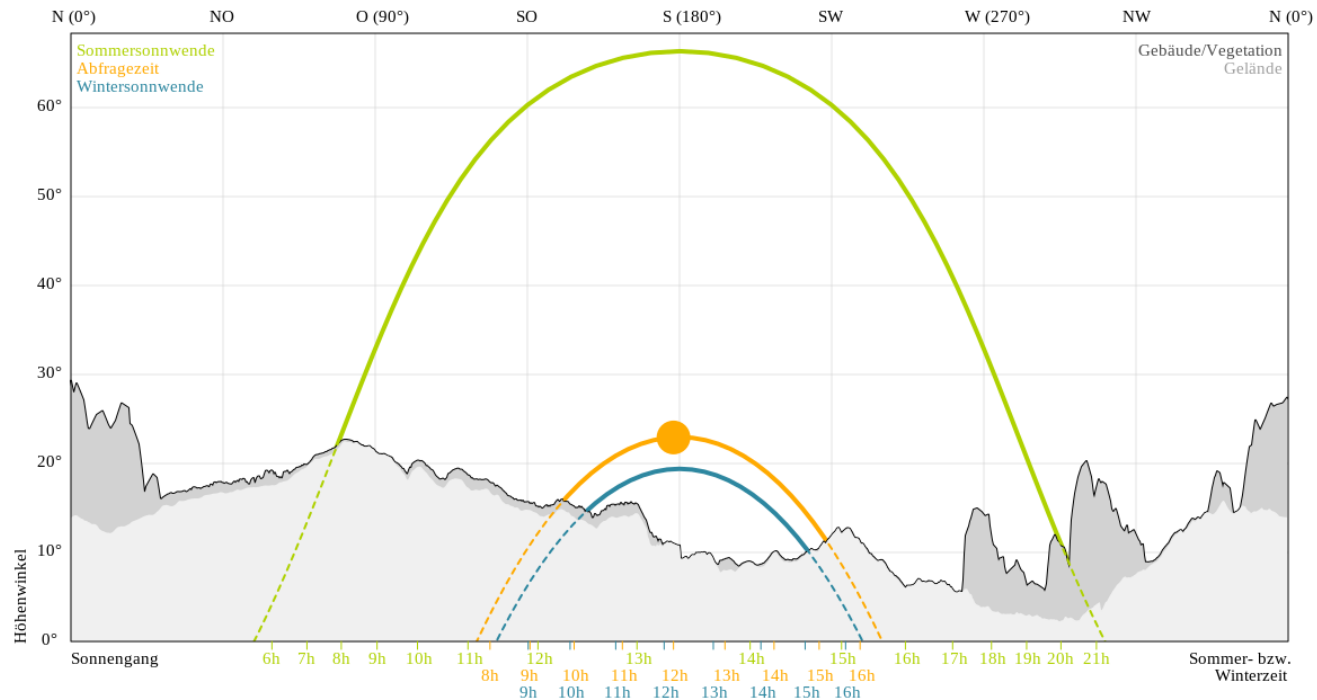
Abfragekoordinaten (EPSG:31254): -41390.00, 228170.00

Abfragehöhe (m): 546.6 (+2.0)

Abfragezeit: 21.11.2022, 12:00 Uhr (Sonnenaufgang 9:45 Uhr, Sonnenuntergang 15:11 Uhr)

Datengrundlage: Laserscanning Höhenmodell 2021 - geoland.at

Befliegungsjahr im Abfragepunkt: 2017



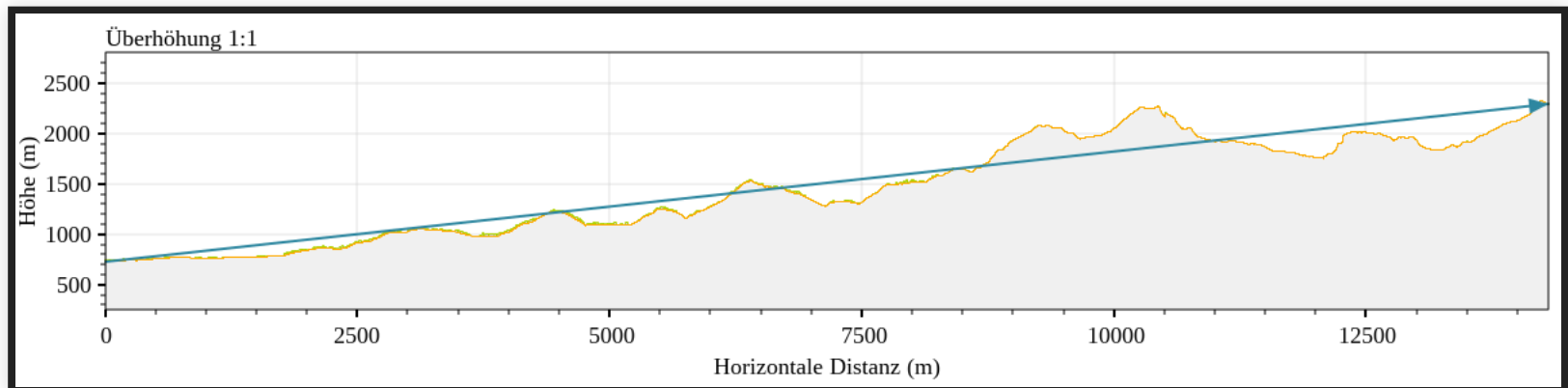
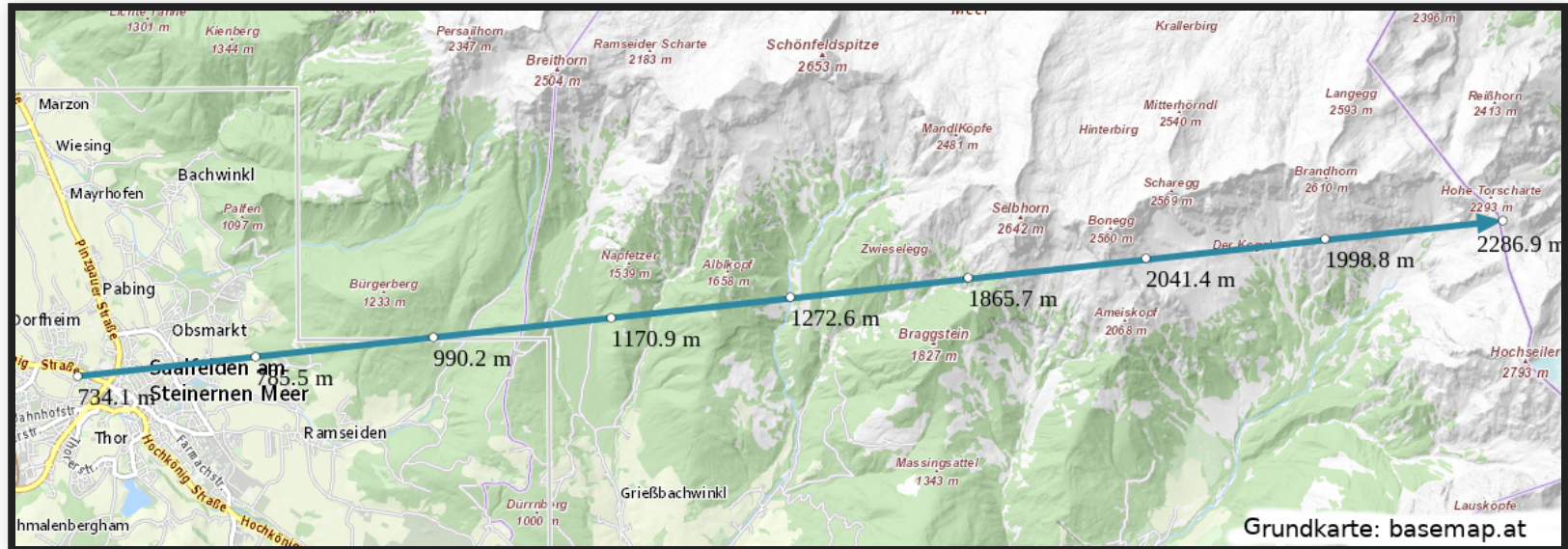
Dokumentation

Höhenprofil

```
https://voibos.rechenraum.com/voibos/voibos?  
name=profilservice&  
Beschriftung=Beispiel&  
Startkoordinate=12.83935,47.42960&  
Zielkoordinate=13.027871,47.443578&  
CRS=4326&  
Ueberhoehung=1&  
Stuetzpunktabstand=2&  
Hintergrund=geolandbasemap&  
Output=Profil,Lage
```

Dokumentation

Höhenprofil



Links

- Folien
- meggsimum GeoServer Instanz
- meggsimum Webseite
- Vortrag in FOSSGIS Pretalx
- Kontakt: jakob@meggsimum.de

Impressum

Autor

Jakob Miksch

meggsimum - Büro für Geoinformatik (Christian Mayer)

Schillerstraße 2a

67122 Mutterstadt

info@meggsimum.de



Lizenz

Diese Folien sind unter **CC BY-SA** veröffentlicht.

Vortragsfolien, PDF-Version, git repository