



QGIS Web Client 2 (QWC2) - Neues aus dem Projekt

Sandro Mani
Sourcepole AG, Zürich
www.sourcepole.ch



SOURCEPOLE
Linux & Open Source Solutions



- › Einführung zum QWC2
- › Einführung zu den qwc-services
- › Neues aus dem Projekt



- **QWC2 ist die offizielle Webanwendung zum Publizieren von OGIS Projekte im Web**
 - Erster öffentlicher Release im 2016
 - Moderne, responsive UI
 - Fokus auf Benutzerfreundlichkeit
 - Plugin Architektur, einfach erweiterbar
 - State-of-the-Art Technologien: ReactJS, OpenLayers 7





OGIS Web Client 2

The screenshot displays the QWC Demo web client interface. At the top, there is a search bar with the text "Search place or add map..." and a "QWC Demo" logo. Below the search bar, a "Toggle API demo" button is visible. The main area is a satellite map of Europe with orange outlines representing country borders. On the right side, a "Layers & Legend" panel is open, showing a list of layers: "Demo", "Edit Demo", "Edit Points", "Edit Lines", "Edit Polygons", "Geographic lines", "Country names", "States and Provinces", and "Countries". Each layer has a checkbox and a trash icon. Below the layers list, there are options for "Show layer map tips", "Compare top layer", and "Import layer". At the bottom of the map, there is a status bar with coordinates (1929481 5892447), projection (WGS 84 / Pseudo Mercator), and scale (1 : 25000000). A "Demo attribution" link is also present. On the left side, a sidebar menu is open, showing options like "Theme", "Layers & Legend", "Layer catalog", "Share Link", "Bookmarks", "Map Tools", "Print", "Help", and "Login".



QGIS Web Client 2 - Übersicht



- QGIS Project
- Layers
 - Styles
 - Print templates

- 1 Karte in QGIS gestalten
- 2 Mit QGIS Server publizieren
- 3 Im Web mit QWC2 betrachten



OGIS Web Client 2 - Kernfunktionalitäten

- Themenswitcher (OGIS Projekte)
- Layerbaum
- Objektinformation
- Suche mit konfigurierbaren Providers
- Messwerkzeug
- Redlining (skizzieren)
- Teilen von Permalinks
- Geolokalisierung
- PDF Druck
- Ebenen vergleichen
- Externe Ebenen importieren (WMS, WFS, WMTS, KML)
- Javascript API



QWC2 und qwc-services

- › QWC2 kann als eigenständigen Viewer vor QGIS Server betrieben werden
 - › Statische JS/HTML/CSS Anwendung
 - › <https://github.com/qgis/qwc2-demo-app>
- › Oder er kann mit zusätzlichen Dienste erweitert werden

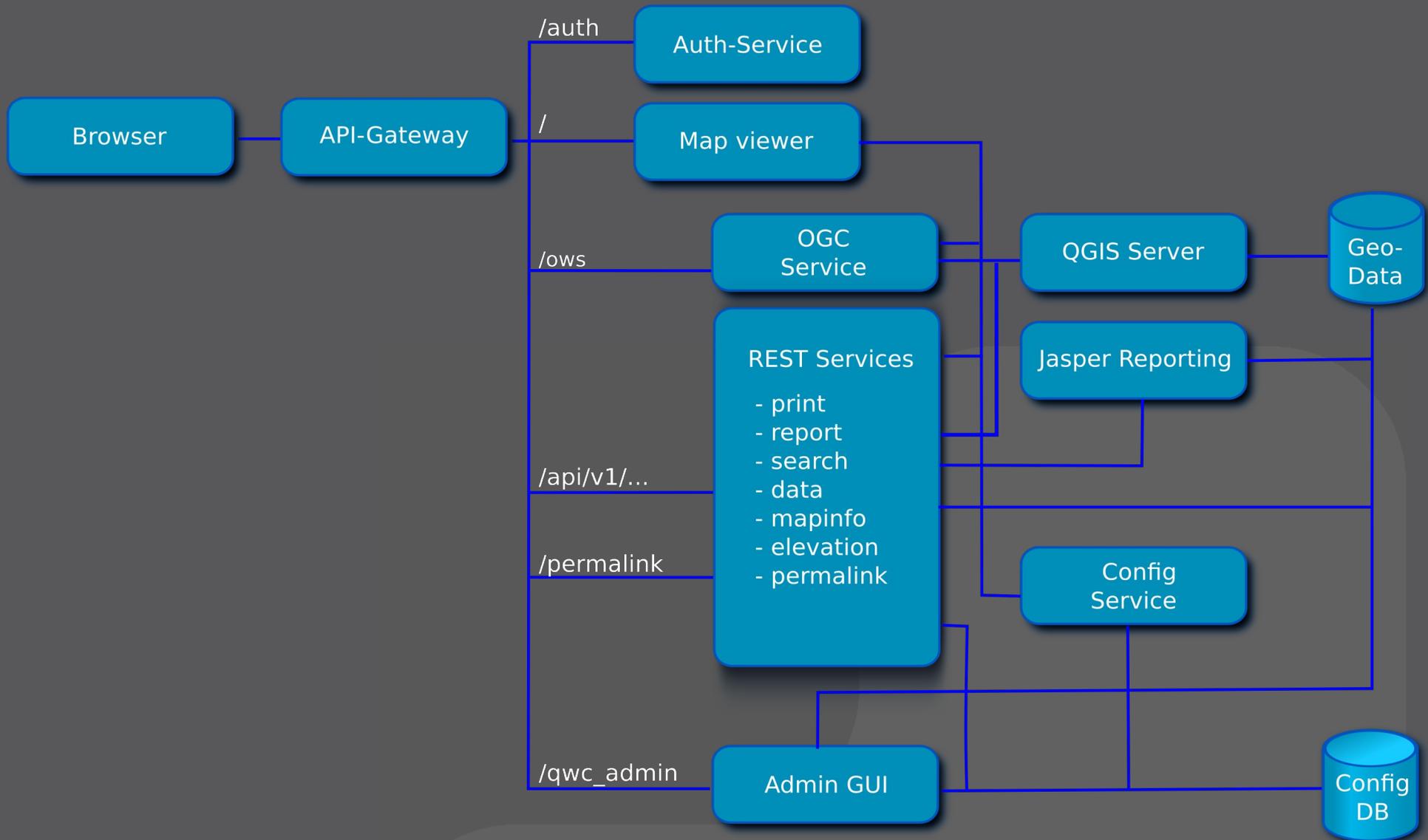
› **qwc-services**



- › **Ecosystem von Mikrodienste zur Erweiterung von QWC2**
 - › Sammlung von schwach gekoppelten Dienste
 - › Kommunikation zwischen den Diensten über HTTP/REST
 - › Hauptsächlich Python-Flask Dienste
- › **Einfaches deployment als Docker-Containers**
 - › <https://github.com/qwc-services/qwc-docker>
- › **Oder personalisiertes Setup, z.B. mit Apache + mod WSGI**



qwc-services - Architektur





- **Verfügbare Funktionalitäten durch Erweiterung von QWC2 mit qwc-services**
 - Benutzer- und Ressourcenverwaltung
 - Authentifizierung (DB, LDAP, ...)
 - Volltext-Suche
 - Editieren
 - Kompakte Permalinks, Lesezeichen
 - Höhenprofil
 - Personalisierte Vorlagen für Objektinfo
 - Dokumente (Jasper)
 - Mapinfo Popup



qwc-services · Kerndienste

› QWC OGC service

- › Leitet Anfragen an den QGIS-Server weiter und erzwingt dabei Berechtigungen

› QWC map viewer

- › Erzwingt Berechtigungen für die QWC2-Themen und die Viewer-Konfiguration

› QWC Admin GUI

- › Backend für das Verwalten von Benutzer- und Ressourcenrechte (Karten, Ebenen, Edit-Datensätze, ...)

› QWC Config Generator

- › Kombiniert Themen und Berechtigungen in statische Dienstkonfig.-Dateien



- › **Modulare Authentifizierungsdienste**
 - › Issue JWT tokens
- › **QWC DB Auth**
 - › Authentifizierung mit Benutzer in DB
- › **QWC LDAP Auth**
 - › Authentifizierung mit LDAP/Active Directory
- › **und weitere (i.e. Kerberos, SAML)**



qwc-services · Einige weitere Dienste

- **QWC Data Service**
 - Ermöglicht Lese-/Schreibzugriff auf konfigurierte Datensätze in einer PostGIS-DB
 - Backend for die QWC2 Editierfunktion
- **QWC Permalink Service**
 - Speichert und löst kompakte Permalinks auf
- **QWC Fulltext Search Service**
 - Facettierte Volltextsuche mit Apache Solr
- **QWC MapInfo Service**
 - Personalisierte, ebenenunabhängige Informationen (Kontext-Menü)



Docker / Kubernetes

› Docker

- › Ein Dienst per Container
- › Ausführung in kontrollierter Umgebung
- › Einfach zu erstellen und zu verwalten
- › Verteilung über eine zentrale Registry
- › <https://www.docker.com/>

› Kubernetes

- › System zur Automatisierung der Bereitstellung, Skalierung und Verwaltung von Containeranwendungen
- › <https://kubernetes.io/>

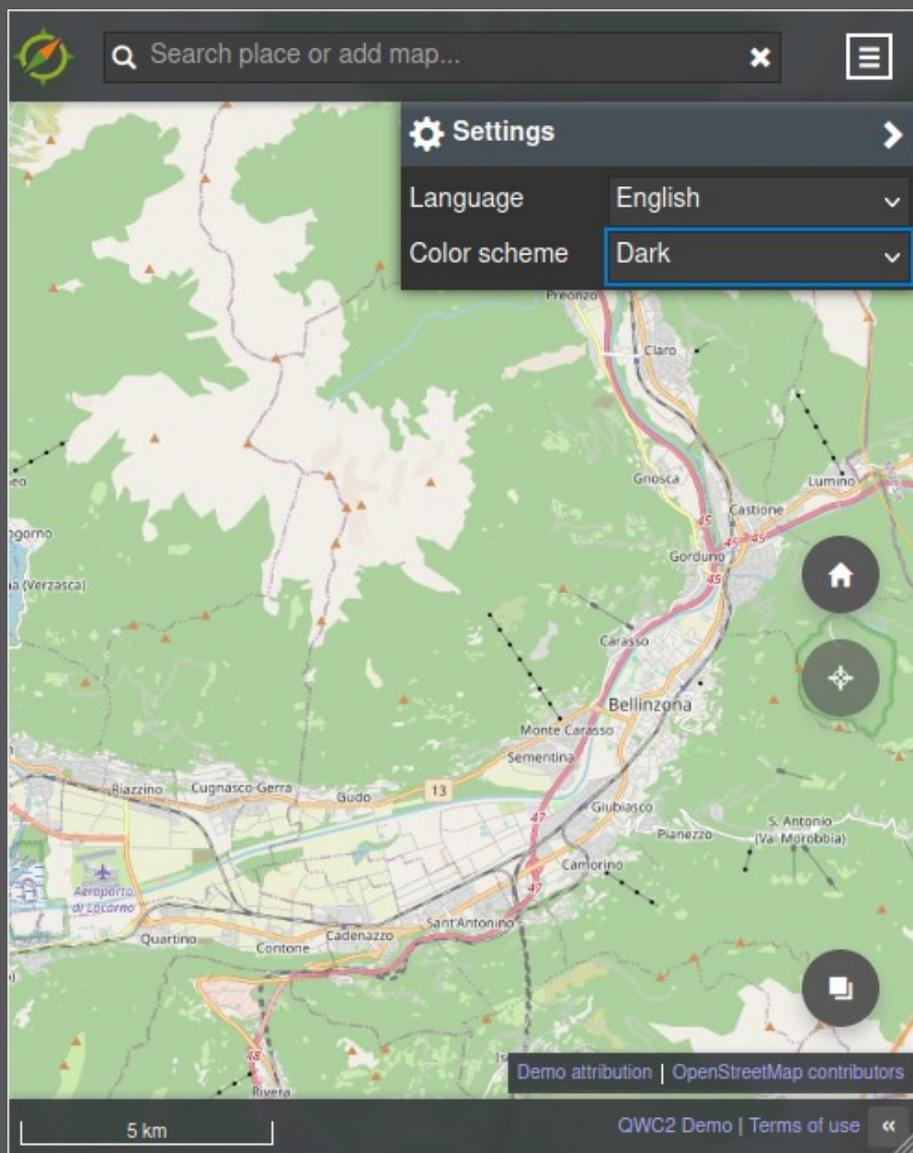


Neues aus dem Projekt

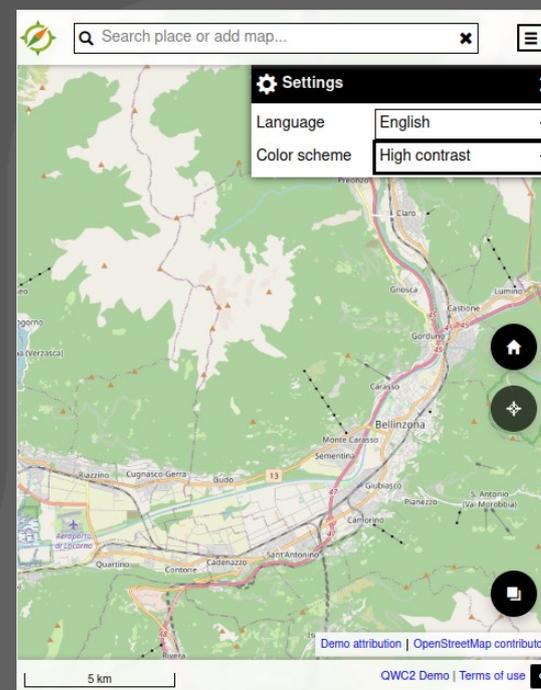


QWC2 News – Long-Term Release

- **Release mit einjähriger Wartung**
 - Stabile Grundlage
 - Keine neuen Features
 - Nur Bugfixes
- **Tags: v2023.x-lts**
- **Docker-images auf dockerhub verfügbar**
- **Aktuellste Releases wie bisher**
 - Tags: vYYYY-MM-DD



- **Frei definierbare Farbschemen über CSS Variablen**
- **Zur Laufzeit umstellbar**





QWC2 News – Time Manager

Search place or add map...

Time manager

Temporal navigation

Markers:

Timescale (non-linearity): 1.00

01	10.06.2007	18.03.2013	25.12.2018	03.10.2024
	07:56:17	12:52:34	18:48:51	01:45:08

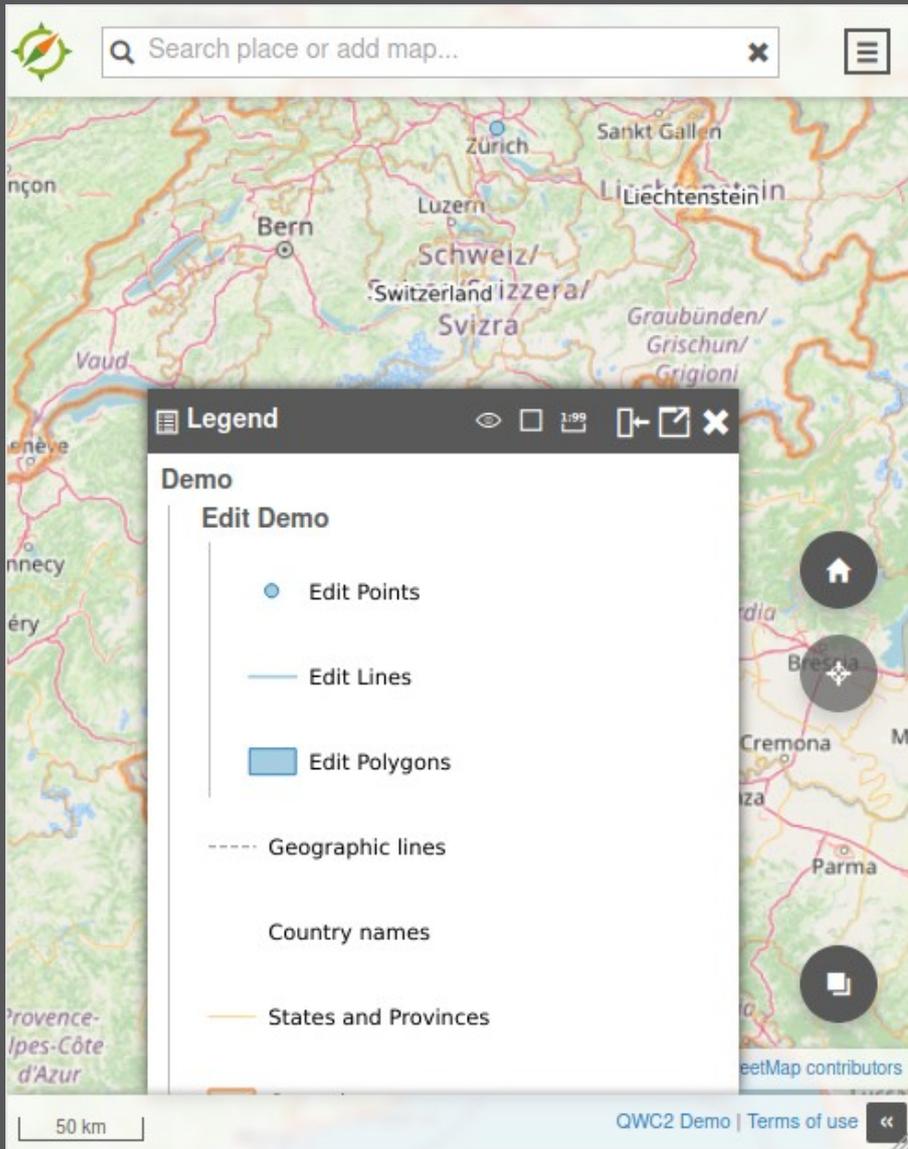
obs_areas Grouping attribute: None Classification attribute: None

area4
area5
area7

20 km

QWC2 Demo | Terms of use

- Neue Objektansicht im Zeitstrahl
 - Gruppierung
 - Klassifizierung
- Fixe und unentliche Zeitachse
- Nicht-lineare Zeitachse



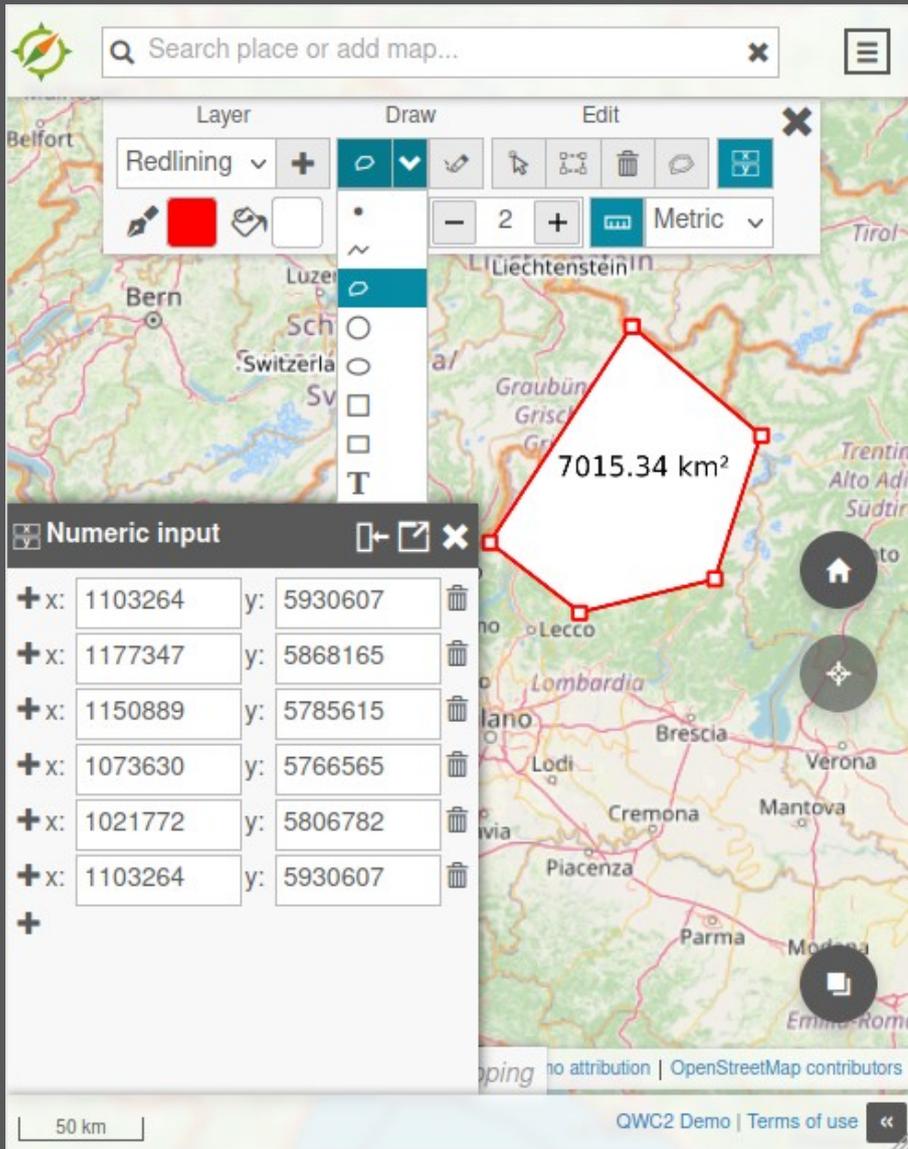
➤ Kartenlegende

➤ Sichtbarkeit:

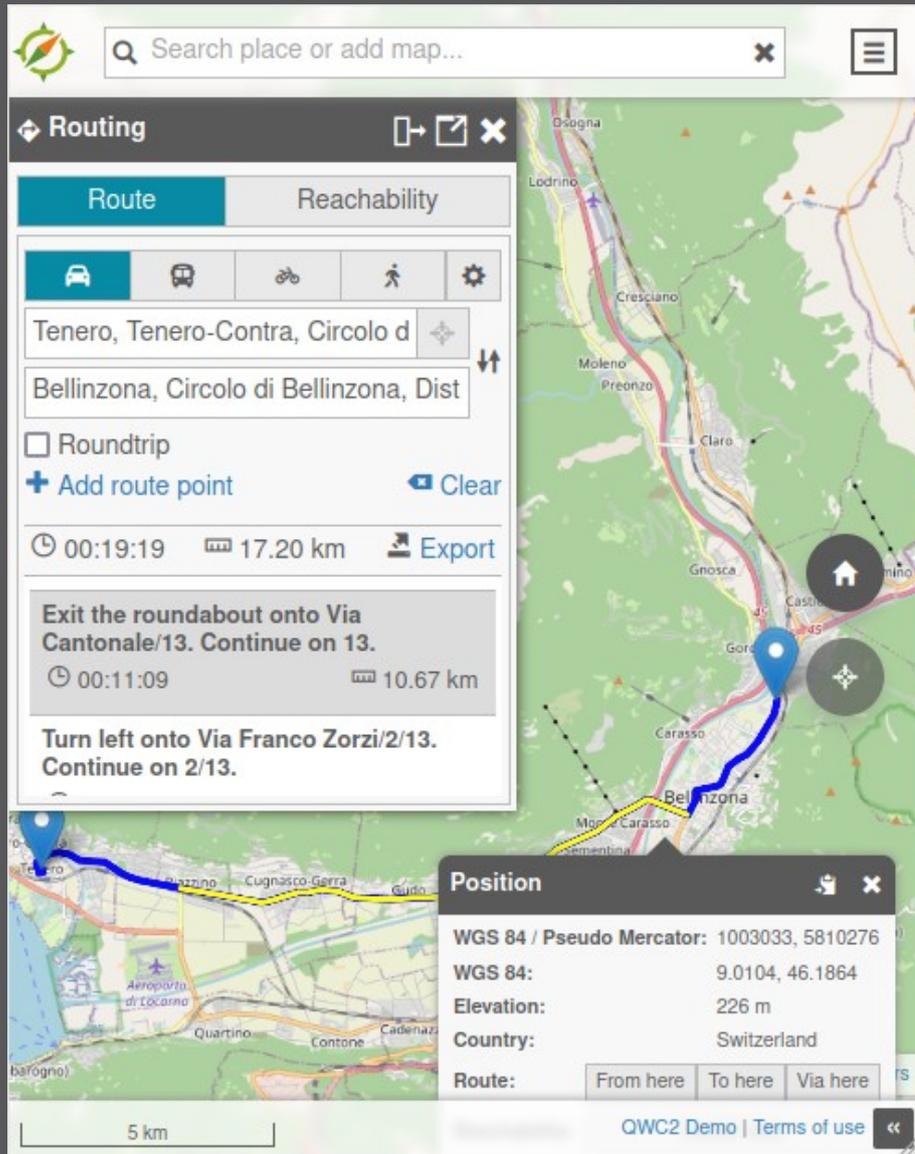
➤ Aktive Ebenen

➤ Im aktuellen Ausschnitt

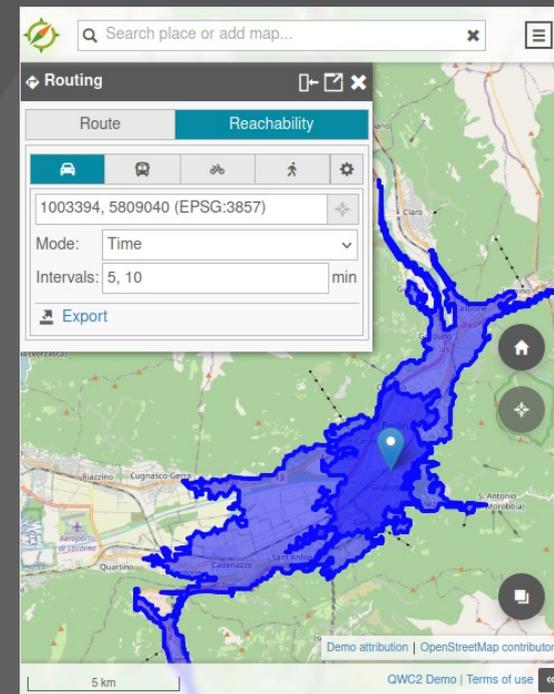
➤ Im aktuellen Massstab



- **Zusätzliche Geometrien**
 - Rechteck
 - Quadrat
 - Ellipse
- **Transformieren**
 - Skalieren
 - Rotieren
- **Bemassung**
- **Numerische Eingabe**



- Routing und Erreichbarkeit
- Default Vallhalla Backend
 - Generisches Interface für andere Routing-Engines





- › Konfigurierbare Tastenkürzel
- › Tastaturnavigation in der Karte und im Hauptmenü
- › Hochkontrast Farbschema

```
"name": "TopBar",
"cfg": {
  "menuItems": [
    {"key": "ThemeSwitcher", "icon": "themes", "shortcut": "alt+shift+t"},
    {"key": "LayerTree", "icon": "layers", "shortcut": "alt+shift+l"},
    {"key": "MapLegend", "icon": "list-alt"},
    {"key": "LayerCatalog", "icon": "catalog", "shortcut": "alt+shift+c"},
    {"key": "Share", "icon": "share", "shortcut": "alt+shift+s"},
    {"key": "Bookmark", "icon": "bookmark", "shortcut": "alt+shift+b"},
    {"key": "Tools", "icon": "tools", "subitems": [
      {"key": "Identify", "icon": "identify_region", "mode": "Region"},
      {"key": "TimeManager", "icon": "clock"}
    ]
  ]
}
```



The screenshot shows the QWC2 interface with a search bar at the top. Below it is the 'Feature Info' panel, which displays two feature entries:

obs_points: Name9	
id	9
name	Name9
startdate	2005-06-10 00:00:00 (UTC)
enddate	2018-05-04 00:00:00 (UTC)
type	4
observer	gr

obs_points: Name13	
id	13
name	Name13
startdate	2008-07-22 00:00:00 (UTC)
enddate	2010-12-07 00:00:00 (UTC)
type	3
observer	wv

Below the feature info, there is an 'Export format' dropdown menu with the following options: json, geojson, CSV, CSV+ZIP, and Object IDs (which is selected). An 'Export' button is visible next to the dropdown. The interface also includes a search bar, a map, and a scale bar (50 km).

➤ Export nach Zwischenablage

➤ Eigene Exportfunktionen

➤ appConfig.js => IdentifyPlugin => customExporters

```
export const customExporters = [
  {
    id: "objectids",
    title: "Object IDs",
    allowClipboard: true,
    export: (features, callback) => {
      const data = Object.values(features).flat().map(feature => feature.id);
      const json = JSON.stringify(data);
      callback({
        data: json, type: "application/geojson", filename: "featureids.json"
      });
    }
  }
];
```



› WMS Basic Auth

- › Geschützte WMS vom OGC Service in Drittapplikationen beziehen (z.B. QGIS)
- › Übergabe von User + Pass. per Basic Auth
- › Prüfung gegen Backend-Service (qwc-db-auth, ...)

› OpenID Connect Authentifizierung

- › <https://github.com/qwc-services/qwc-oidc-auth>

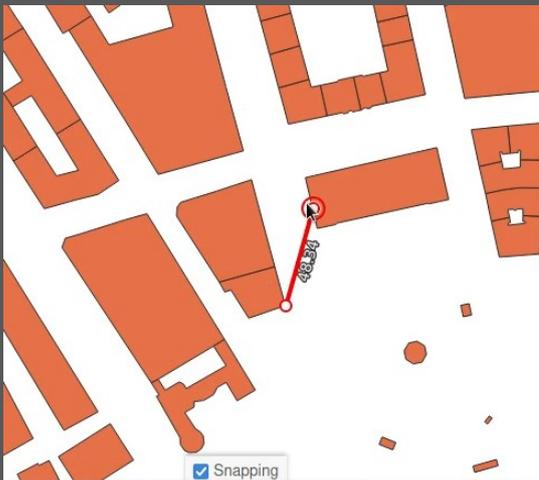


- **Übersichtlichere Pluginkonfiguration durch gemeinsame Basiskonfiguration**
 - Selektives Überschreiben / Erweitern für mobile/desktop

```
],
"plugins": [
  "common": [ ...
  ],
  "mobile": [
    {
      "name": "BottomBar",
      "cfg": {
        "displayCoordinates": false,
        "displayScales": false
      }
    }
  ],
  "desktop": [ ...
  ]
]
```

➤ Weitere Neuerungen des letzten Jahres:

- Snapping
- Attributtabelle
- Objektformular



Attribute Table

Layer: Fluffy animals

	id	name	age	birthdate	size	hungry	sleepy	remarks	Animal	picture
<input type="checkbox"/>	2	Minni	0	04 / 25 / 2011	5	<input type="checkbox"/>	<input type="checkbox"/>		Cat	minni.jpg
<input type="checkbox"/>	1	Mily	2	02 / 13 / 2019	6	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Selec	11.jpg
<input type="checkbox"/>	3	asd	1	mm / dd / yyyy	6	<input type="checkbox"/>	<input type="checkbox"/>		Cat	Browse... ..

1 | 50 per page | id

Feature form

Fluffy animals: Minni

Fluffy animals: Mily

Fluffy animals: Minni

Picture

Age: 0

Birthdate: 04 / 25 / 2011

Gender: Male Female

Size: meters

Flags: Hungry Sleepy



› Homepage

- › <https://qwc-services.github.io/>

› Repositories

- › <https://github.com/qgis/qwc2-demo-app>

- › <https://github.com/qwc-services/qwc-docker>

› Beispiele

- › <https://qwc2.sourcepole.ch>

- › <https://qgiscloud.com>

- › <https://map.geo.gl.ch>

- › <https://geo.so.ch/map/>



Danke!



Sandro Mani