WORLD HEAT FLOW DATABASE PROJECT

Analysis of free web mapping libraries

As decision support tool for the **World Heat Flow Database Project**

Authors: Nikolas Ott ORCiD[1]*, Stephan Mäs ORCiD[1]

Publication date: July 2023

Summary

Further development of front-end technologies makes client-side web mapping applications a promising way of providing (interactive explorable) geodata to the global community. Which libraries address this topic and how the implementation of essential web GIS functionalities can be done using LeafLet, MapLibre or OpenLayers are questions analysed in this document. Therefore, internet research has been done collecting already existing solutions making use of the respective functionalities of the libraries. As a result, links to implementation examples, plugins or references in the API documentations have been gathered in a functionality matrix.
The information gained by this research is used for developing and implementing an interactive web mapping application as part of the World Heat Flow Database Project. Beside this, the document offers an overview for people who are new to this topic, provides code examples for common web GIS features and makes the readers familiar with topic-specific vocabulary and terms.

[1] Chair of Geoinformatics, Technische Universität Dresden
* Corresponding author email: nikolas.ott@tu-dresden.de

Contents

Introduction

To raise the accessibility of geographic data, modern open-source web mapping libraries like Leaflet, MapBox, MapLibre or OpenLayers provide powerful visualization and analysis tools for geodata. This is enabled thanks to the advancement of technologies like HTML5, JavaScript and WebGL, which are nowadays supported by standard browsers (Zunino et al. 2020).

A comparison of the provided features of the available libraries is of interest for the *World Heat Flow Database Project* where the *Geoforschungszentrum (GFZ) Potsdam* and the *Chair of Geoinformatics* of the *Technische Universität Dresden* develop and implement a new research data infrastructure for the IHFC Global Heat Flow Database. The resulting one-stop shop will offer comprehensive information about heat flow and related data like publications, projects, and researchers. To make this a user-oriented web service, an adjusted set of exploration and visualization tools, suited for heat flow data will be provided. Therefore, an interactive web map application is going to be integrated.

The analysis of which Web Mapping Libraries exist and what functionalities they support has been summarized in this document. The results obtained in the following were used as a decision support tool for the project.

The structure of the document is divided in three sections:

1. Market analysis of Web Mapping Libraries
2. Functionality check
3. Outlook

Market analysis of Web Mapping Libraries

As an initial step, a market analysis has been performed. Different kinds of sources, like the internet or journals, have been gathered to get an overview of already existing solutions. The use of well documented and tested existing code and therefore the avoidance of duplications fosters good practice.

Ammann et al. (2022) wrote a paper where they give an overview of open-source map renderers for mobile, desktop and web applications. Because the web renderers are of special interest in our case, Table 1 shows the relevant subset of their review containing open-source and source-available map renderers written in JavaScript/TypeScript. Additionally, Table 1 is extended by libraries found in connection with the research conducted here. Each renderer is linked to its repository and described via the supported input formats, the license and the GitHub stars, as a first rough indicator of popularity.

Table 1: Subset of web map renderers according to (Ammann et al. 2022)

Repository	Input formats			License	GitHub Stars [K][2]
	Raster tiles	Vector tiles	Other		
CesiumGS/cesium	✓		✓	Apache-2.0	9.5
heremaps/harp.gl	✓	✓	✓	Apache-2.0	1.2
hijiangtao/glmaps	✓		✓	MIT	0.347
iTowns/itowns	✓	✓	✓	Cecill-B/MIT	0.773
Leaflet/Leaflet	✓	✓	✓	BSD-2	36
mapbox/mapbox-gl-js	✓	✓	✓	Source-avail.	9.1
maplibre/maplibre-gl-js	✓	✓	✓	BSD-3	3.7
NASA-AMMOS/3DTilesRendererJS	✓	✓	✓	Apache-2.0	0.933
openlayers/openlayers	✓	✓	✓	BSD-2	9.5
potree/potree			✓	BSD-2	3.3
felixpalmer/procedural-gl-js	✓		✓	MPL-2.0	1.2
protomaps/protomaps.js	✓	✓	✓	BSD-3	0.366
tangrams/tangram	✓	✓	✓	MIT	2.1
visgl/deck.gl	✓	✓	✓	MIT	10.4
OpenGeoscience/geojs	?	?	?	Apache-2.0	0.378
geoext/geoext	?	?	?	Unknown	0.132
maptalks/maptalks.js	?	?	?	BSD-3-Clause license	3.8

[ ] = in addition to (Ammann et al. 2022)

[2] As on 03.11.2022

Appropriate libraries

In order to meet the requirements of this project, some criteria have to be fulfilled by the web renderers. The criteria are listed in the following:

- Library audience is geo specific:
  - Designed for geodata
  - Handling of raster and vector tiles
  - Provide visualization and analysis tools
- Reliable community for:
  - Support through other users
  - Finding solutions for common issues
  - Simplify error handling
- Usage is at no charge
- Open source

After filtering the map renderers by the above-described criteria, three libraries remain. They will be introduced shortly by their characteristics in the following sub-section.

Leaflet

First released in May 2011, it has now reached the latest version 1.9[3]. Leaflet describes itself as a lightweight library which nevertheless contains the most common mapping features[4]. The core functionality can be extended by adding thematic third-party plugins to it[5].

MapLibre

When Mapbox GL JS changed their license from open source to closed source in December 2020, MapLibre was born. As a community project, the MapLibre coalition forked the last open-source version of Mapbox and made it their goal to maintain and further develop it from then on[6]. Right now, version 2.4 has been released. The main focus is on performance, usability and a low-threshold access to development[7].

OpenLayers

OpenLayers released the first version in June 2006[8] and has become a serious open-source alternative to Google Maps API these days. Since then, the project has advanced more and more and now reached version 7.2.2. OpenLayers provides a full-featured

[3] https://en.wikipedia.org/wiki/Leaflet_(software) accessed: 01.02.2023
[4] https://leafletjs.com/ accessed: 01.02.2023
[5] https://leafletjs.com/plugins.html accessed: 01.02.2023
[6] https://www.maptiler.com/news/2021/01/mapbox-gl-open-source-fork/ accessed: 01.02.2023
[7] https://pretalx.com/fossgis2021/talk/FNFSMF/ accessed: 02.02.2023
[8] https://en.wikipedia.org/wiki/OpenLayers accessed: 02.02.2023

mapping library which enables the implementation of complex web mapping applications. [9]

Functionality check

In this section, the above selected web mapping libraries are analysed more specifically in terms of functionality.

Therefore, core functionalities have been listed including essential web GIS features, like handling different kinds of data types, selecting features by location and more. Already existing solutions have been gathered from the internet. The links to implementation examples, plugins, tutorials or references in the API documentations are gathered in *Table 2*. The three web mapping libraries are set as columns and each functionality is represented by a row. One field can contain zero to multiple entries separated by bullets. If no solutions outside of the box can be found, a first workflow will be provided as a numbered list.

[9] https://dev.to/camptocamp-geo/the-3-best-open-source-web-mapping-libraries-57o7 accessed: 02.02.2023

Table 2: required functionalities and possible solutions according to the library (functionality matric). All Links were last opened on 24.04.2023

Functionalities		Leaflet v1.9	MapLibre v2.4.0	OpenLayers v7.3.0
General	**Support WebGL**	Only by plugins: • Leaflet.TileLayer.GL • Leaflet.TileLayer.GLColorScale • Leaflet.TileLayer.GLOperations	✓	• ol/webgl • WebGLTileLayer • WebGLArrayBuffer • WebGLRenderTarget • WebGLPostProcessingPass • WebGLTileLayerRenderer • WebGLPointsLayerRenderer
	Multiple map languages	• leaflet-extras/leaflet-providers • Map internationalization	• Mapbox GL Language	• Map internationalization
Data	**Basemap**	• Basemap formats • leaflet-extras/leaflet-providers	• MapLibre/style/source • Add a raster tile source	• Ol/source • ol/source/OSM~OSM • Stamen Tiles
	Vector file formats	• Overlay data formats	• GeoJSON	• Ol/format/Feature
	Raster file formats	• https://leafletjs.com/plugins.html	• ImageSource	• ol/source/Raster~RasterSource
	Vector tiles	• Vector tiles (mapbox/geojson-vt → returns MVT)	• vector-tiles (Mapbox Vector Tile format, mapbox/geojson-vt → returns MVT)	• MVT (mapbox/geojson-vt → returns MVT)
	Raster tiles	• Tile & image layers • Tile/image display	• raster	• ol/layer/Tile~TileLayer
	OGC services	• WMS • WMTS • WFS • OGC API Features → opengeospatial/ogcapi-features	• WMS • WMTS (set URL to TileMatrixSet {z}/{y}/{x}.png)	• WMTS • WMS • WFS

s			• OGC API Features → opengeospatial/ogcapi-features	• OGC API Features → opengeospatial/ogcapi-features
	Projections	• kartena/Proj4Leaflet	• Geography and geometry • Issue: Support rendering in multiple CRS #168	• ol/proj/Projection~Projection • OpenLayers Examples
	Import user data	• Leaflet.FileLayer	• View local GeoJSON	• Drag n drop
	Download features (GeoJSON)	1. Bring features to GeoJSON format and change the input of the following tutorial to the GeoJSON 2. How to create a file and generate a download with Javascript in the Browser (without a server)	1. Bring features to GeoJSON format and change the input of the following tutorial to the GeoJSON 2. How to create a file and generate a download with Javascript in the Browser (without a server)	1. Bring features to GeoJSON format and change the input of the following tutorial to the GeoJSON 2. How to create a file and generate a download with Javascript in the Browser (without a server)
UI interactions	**Layer switcher**	• Layer switching controls • Layer Groups and Layers Control	• MapLibre GL Basemaps Control • mapbox-gl-legend • Maplibre GL - Ebenen ein- und ausblenden	• OpenLayers LayerSwitcher
	Data driven feature visualization	• lizardtechblog/Leaflet.OpacityControls • Leaflet: Change color onClick	• Change a layer's color with buttons • Dayjournal/maplibre-gl-opacity • Style circles with a data-driven property	• Color Manipulation • Change Tile Layer Style • ol-ext: Color filter on map • Making it look nice
	Graticule	• leaflet.latlng-graticule	• maplibre-grid	• Map Graticule
	Time slider	• LeafletSlider • svitkin/leaflet-timeline-slider	• Create a time slider	• Animating meteorite impacts → from CSV • Creating a Timeseries with GeoServer and Open Layers → from TIF (raster)

				• ol-kit/src/TimeSlider/
	Cluster	• Leaflet/Leaflet.markercluster • Clustering/Decluttering	• Create and style clusters	• Clustered Features
	Show coordinates of cursor	• Leaflet.MousePosition	• Get coordinates of the mouse pointer	• Mouse Position
	Popup	1. DivOverlay Popup 2. Using GeoJSON with Leaflet	3. Popup 4. Examples of Popup	5. Popup
	Export map/ view	• Print/export	• maplibre-gl-export	• Map Export
Select by location	**Geocoding (Search by location)**	• Leaflet Geocoding	• Geocode with Nominatim	• OpenLayers Control Geocoder
	Select feature	• Using GeoJSON with Leaflet	• Get features under the mouse pointer	• Select Features
	Select features within circle	1. L.Draw.Circle 2. Turfjs buffer → returns polygon 3. Turfjs pointsWithinPolygon	1. iamanvesh/mapbox-gl-draw-circle 2. Turfjs buffer → returns polygon 3. Turfjs pointsWithinPolygon	1. Draw Features → Circle 2. Turfjs buffer → returns polygon 3. - Turfjs pointsWithinPolygon - OpenLayers demo for selecting by polygon, circle, square and rectangle
	Select features within bounding box	1. L.Draw.Rectangle 2. Turfjs pointsWithinPolygon	1. Mapbox GL Draw Rectangle Mode 2. Turfjs pointsWithinPolygon	1. Draw Shapes → Box 2. - Turfjs pointsWithinPolygon - OpenLayers demo for selecting by polygon, circle, square and rectangle
	Select features within polygon	1. L.Draw.Polygon 2. Turfjs pointsWithinPolygon	1. Draw a polygon and calculate its area 2. Turfjs pointsWithinPolygon	1. Draw Features → Polygon 2. Turfjs pointsWithinPolygon - OpenLayers demo for selecting by polygon, circle, square and rectangle

Outlook

Table 2 shows, that for each functionality there is mostly one already existing solution to implement it in each of the three libraries. All of them have their strengths and weaknesses. Which library suits best is, in the end, up to the purpose of the project. Some additional third-party libraries have been found during this analysis, dealing with spatial data analysis, coordinate transformation, data manipulation and other related topics. A collection of them can be found in Table 3.

We would be glad to extend the functionality matrix with input from the community. Therefore, you are invited to add further functionalities to the table or extend the already existing collection of links.

Add a new Link to an existing functionality

How could this be done?

The table is also stored in a Git repository (free-web-mapping-libraries) as a Markdown document to make it extendable for other people. Just send a pull request containing the addition. Therefore, a basic template should be satisfied:

⟷ Within one row:	links only related to the corresponding functionality
⬍ Within one column:	links only related to the corresponding library
⟷ + ⬍ Within one field:	links only related to the corresponding functionality and library following the template below

- Add bullet point

```
<li>
   <a href="< link_to_web_source >"><title of repo/article/homepage, …></a>
</li>
```

- Add numbered lists (e.g., ordered workflow)

```
<ol type="1">
   <li>
      <a href="< link_to_web_source >"><title of repo/article/homepage, …></a>
   </li>
</ol>
```

Literature

Ammann, M.; Drabble, A.; Ingensand, J.; Chapuis, B. (2022): MAPLIBRE-RS: TOWARD PORTABLE MAP RENDERERS. In: *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.* XLVIII-4/W1-2022, S. 35–42. DOI: 10.5194/isprs-archives-XLVIII-4-W1-2022-35-2022.

Zunino, Alejandro; Velázquez, Guillermo; Celemín, Juan; Mateos, Cristian; Hirsch, Matías; Rodriguez, Juan (2020): Evaluating the Performance of Three Popular Web Mapping Libraries: A Case Study Using Argentina's Life Quality Index. In: *IJGI* 9 (10), S. 563. DOI: 10.3390/ijgi9100563.

Additional third-party libraries

Table 3: Additional thematic related JavaScript libraries

GitHub Repository	Description	License	GitHub Stars [tsd][1]
Turfjs/turf	"Advanced geospatial analysis for browsers and Node.js"[10]	MIT license	7.5
naturalatlas/node-gdal	"Read and write raster and vector geospatial datasets straight from" Node.js with this native GDAL binding"[11]	Apache-2.0 license	0.519
proj4js/proj4js	"Proj4js is a JavaScript library to transform point coordinates from one coordinate system to another, including datum transformations"[12]	Link to license	1.7
mbloch/mapshaper	"Mapshaper is software for editing Shapefile, GeoJSON, TopoJSON, CSV and several other data formats, written in JavaScript "[13]	Mozilla Public License, v. 2.0	3.1
GeoTIFF/geoblaze	"GeoBlaze is a geospatial raster processing engine written purely in javascript. Powered by geotiffjs, it provides tools to analyze GeoTIFFs"[14]	MIT	0.120
geotiffjs/geotiff.js	"Read (geospatial) metadata and raw array data from a wide	MIT	0.630

[10] https://turfjs.org/
[11] https://github.com/naturalatlas/node-gdal#node-gdal
[12] https://github.com/proj4js/proj4js#proj4js-
[13] https://github.com/mbloch/mapshaper#introduction
[14] https://github.com/GeoTIFF/geoblaze#geoblaze

	variety of different (Geo)TIFF files types"[15]		
MapLibre/Plugins	"Most plugins are designed to work with Mapbox GL JS, but should also work with MapLibre GL JS."	Depends on Plugin	-
Openlayers/Extentions	"The libraries below provide additional functionality by extending OpenLayers or integrating well with it."	Depends on Plugin	-
Leaflet Plugins database	"While Leaflet is meant to be as lightweight as possible, and focuses on a core set of features, an easy way to extend its functionality is to use third-party plugins. Thanks to the awesome community behind Leaflet, there are literally hundreds of nice plugins to choose from."	Depends on Plugin	-
bjornharrtell/jsts	"JSTS is an ECMAScript library of spatial predicates and functions for processing geometry conforming to the Simple Features Specification for SQL published by the Open Geospatial Consortium"	Eclipse Public License, Version 1.0 (EPL-1.0)	1.3
maputnik/editor	"A free and open visual editor for the Mapbox GL styles targeted at developers and map designers."	MIT	1.5
bpostlethwaite/colormap	-	MIT	0.182

[15] https://github.com/geotiffjs/geotiff.js#geotiffjs

mbloch/mapshaper	• "Mapshaper is software for editing Shapefile, GeoJSON, TopoJSON, CSV and several other data formats, written in JavaScript" • Convert between popular geo file formats (SHP, GeoJSON, TopoJSON, DBF, CSV)	Mozilla Public License, v. 2.0	3.1
caseycesari/GeoJSON.js	"Turn your geo data into GeoJSON."	MIT	0.255
mapbox/tippecanoe	"Builds vector tilesets from large (or small) collections of GeoJSON, Geobuf, or CSV features, like these."	BSD-2	2.2
Danfo.js	"Danfo.js is an open-source, JavaScript library providing high-performance, intuitive, and easy-to-use data structures for manipulating and processing structured data."	MIT	3.9
gka/chroma.js	"Chroma.js is a tiny small-ish zero-dependency JavaScript library (13.5kB) for all kinds of color conversions and color scales."		9.3
adobe/leonardo	"Authoring adaptive color palettes for generating color based on a desired contrast ratio."	Apache-2.0 license	1.5