



Neues und Unbekanntes in OpenLayers

Marc Jansen

Andreas Hocevar

FOSSGIS 2022, Online-Event, 10.03.2022

Gliederung

- Über / Meta 
- Beispiele 
- Ausblick 

über / Meta



Marc Jansen



✉ jansen@terrestris.de

👤 [@marcjansen](https://www.linkedin.com/in/marcjansen)

🐦 [@selectoid](https://twitter.com/selectoid)



- Geschäftsführer **terrestris** & **mundialis**
- Kernentwickler & PSC OpenLayers
- GeoExt, SHOGun, GeoStyler ...
- Sprecher & Trainer national & international
- OSGeo Foundation Charter Member

Andreas Hocevar

ahocevar geospatial



✉ mail@ahocevar.com
👤 [@ahocevar](#)
🐦 [@ahoce](#)

- Kernentwickler & Maintainer mehrerer FOSSGIS Projekte
- Professioneller OpenLayers Support
- Mitbegründer von [w3geo.at](#)

Was ist OpenLayers?

“

*A high-performance, feature-packed
library for all your mapping needs.*

– openlayers.org, 04. Februar 2022

- OpenSource (BSD)
- JavaScript
- OSGeo Projekt

Grundkonzepte

- Überlagern von Karten aus verschiedenen Quellen
- Navigieren im Kartenfenster
- Interaktion mit den zugrundeliegenden Daten
- Daten- und auflösungsabhängige Darstellung
- Manipulation von Darstellung und Daten

Anwendungsbereiche

- Geographie, Kartographie, Vermessung, GIS, ...
- Photographie, Archivierung
- Medizin, Mikroskopie
- Online Games

Historisches

2006: v2 + OSM = 1. freie Alternative zu Google Maps

2012-2014 v3: Komplettes Rewrite mit neuem API

2014-2017: Krise! Kein Upgradepfad von v2 auf v3

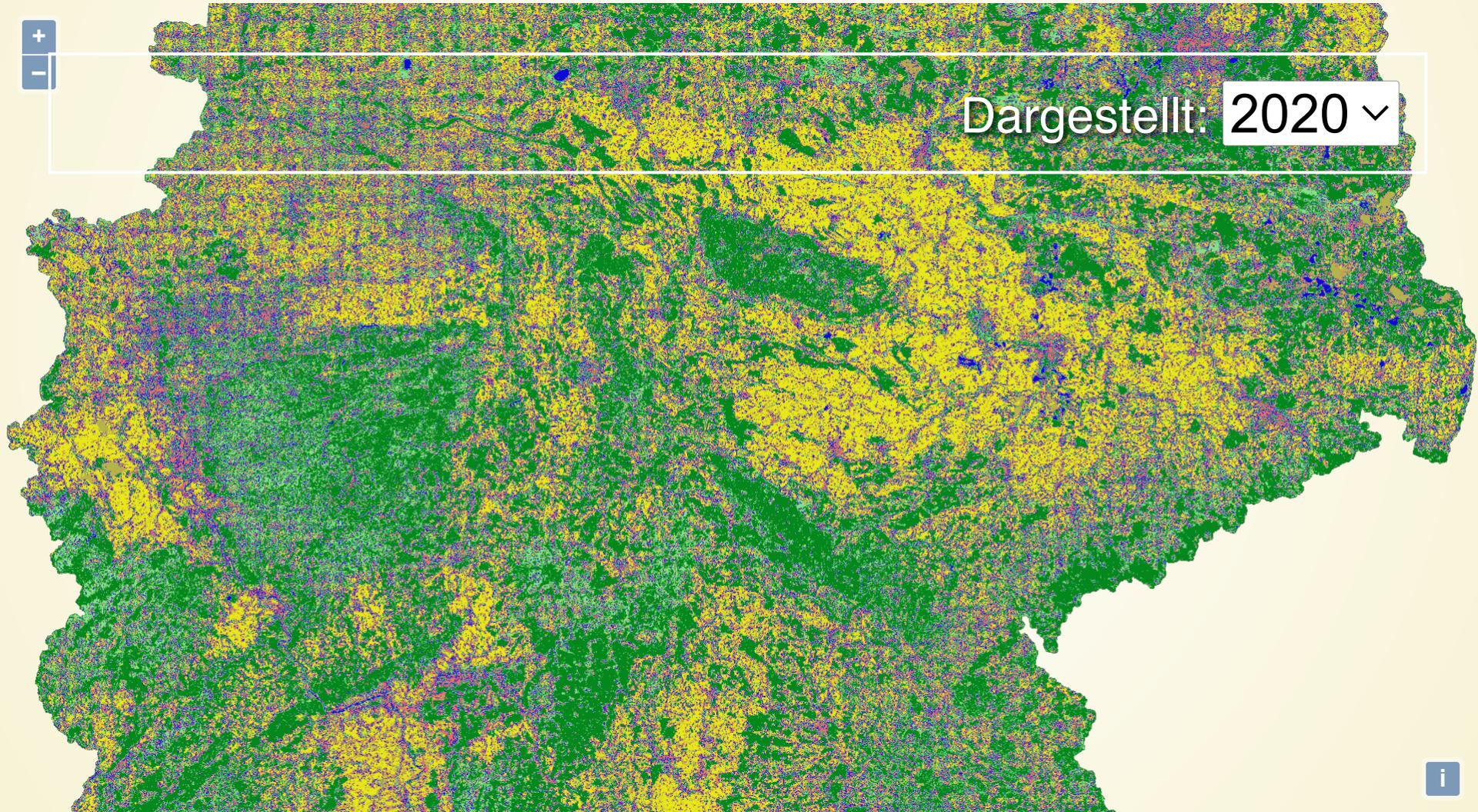
2017 v4: Wir gewinnen wieder Vertrauen und Nutzer

2022 v6.13: Wir sind immer noch da, aktiver denn je!

Beispiele



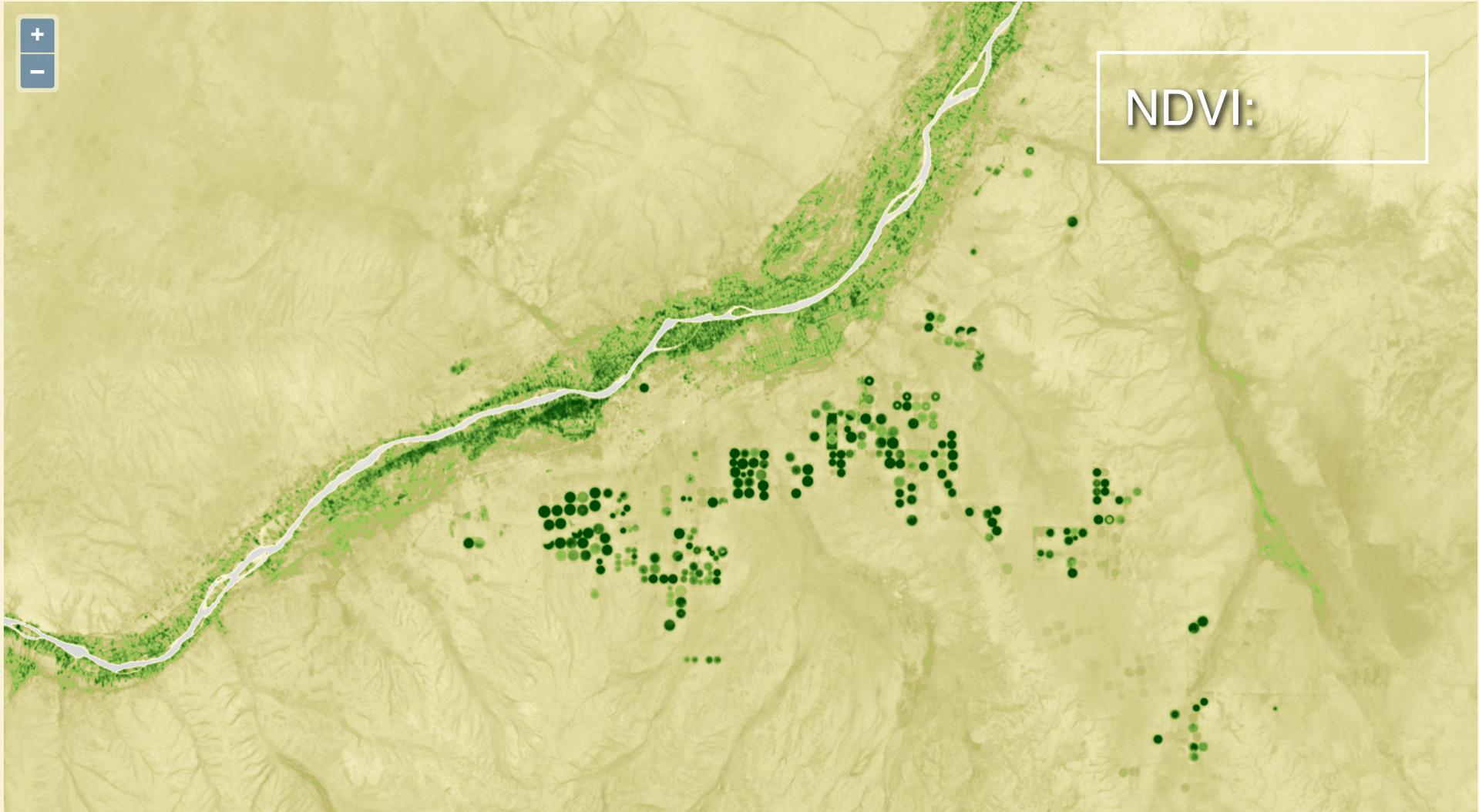
Rechnen mit Pixeln...



Welche Pixel haben sich geändert?

```
{  
  color: [  
    'case',  
    ['==', ['band', 1], ['band', 2]],  
    [0, 0, 0, 0], // equal  
    [255, 165, 0, 0.8] // different  
  ]  
}
```

Layer#getData()



Layer#getData()

Rohdaten abfragen

```
const data = layer.getData(event.pixel);  
if (!data) {  
  return;  
}  
  
const red = data[0];  
const nir = data[1];  
const ndvi = (nir - red) / (nir + red);
```


WebGL Data Tiles

Band Math - Rechnen mit RGB

```
// elevation = -10000 + ((R * 256 * 256 + G * 256 + B) * 0.1)

const elevation = [
  '+',
  -10000,
  ['*',
    ['+',
      ['*', 256 * 256, ['band', 1]],
      ['+', ['*', 256, ['band', 2]],
      ['band', 3]]
    ],
  0.1 * 255,
],
];
```

WebGL Data Tiles

Style Variablen

```
const style = {
  variables: {
    stauziel: 368,
  },
  color: [
    'case',
    ['<=', elevation, ['var', 'stauziel']],
    [139, 212, 255, 1],
    [139, 212, 255, 0],
  ],
}
```

WebGL Data Tiles

Aktualisierung nach User-Interaktion

```
slider.addEventListener('input', () => {  
  layer.updateStyleVariables({  
    stauziel: parseFloat/slider.value)  
  });  
});
```

Multi-Source Layer

Schon mal in der Situation gewesen, für einen Layer je nach Zoom und Ausschnitt unterschiedliche Sources verwenden zu wollen?

Multi-Source Layer



LifeGate 2021
Interaktive phylogenetische Karte

> Über das Projekt

> Unterstützen

> iDiv

> Impressum

> Interaktive Karte

Über das Projekt

LifeGate 2021 ist eine interaktive navigierbare Karte, in der versucht wird, **alle bekannten Arten** (2,6 Mio.) nach ihren phylogenetischen Beziehungen in einer zweidimensionalen Darstellung graphisch anzuordnen.

Die Kartenansicht kann in unterschiedlicher Größe und zwischen folgenden farblich kodierten taxonomischen Stufen erkundet werden:

- Domäne,
- Stamm (*der Ausgangspunkt der Karte*),
- Klasse,
- Ordnung,
- Familie,
- Gattung,
- Art.

Die **Fläche, die eine taxonomische Gruppe einnimmt**, hängt davon ab, wie viele Arten sie umfasst. Taxonomische Gruppen mit vielen enthaltenen Arten sind entsprechend größer in der Darstellung. So befinden sich in der Familie der Ameisen *Formicidae* mehr als doppelt so viele Arten wie in der gesamten Klasse der Säugetiere (*Mammalia*). Die verschiedenen Taxa wurden in einem numerischen Verfahren unter Verwendung selbstähnlicher Voronoi-Polygone nach ihren Verwandtschaftsbeziehungen angeordnet. Taxa mit einer geringeren Anzahl zugeordneter Arten – häufig Relikte phylogenetisch älterer Verzweigungen – befinden sich dabei **am linken unteren Rand** ihrer jeweiligen Zelle. Es werden nur rezente (d. h. keine prähistorisch ausgestorbenen) Arten berücksichtigt.

DE | EN

Multi-Source Layer

STAC Tiled Assets Extension

```
const pyramid = new WebGLTileLayer({
  sources: sourcesFromTileGrid(
    tileGrid,
    ([z, x, y]) => new GeoTIFFSource({
      sources: [ { url: `./data/${z}/${y}/${x}.tif` } ],
    })
  ),
});
```

Multi-Source Layer

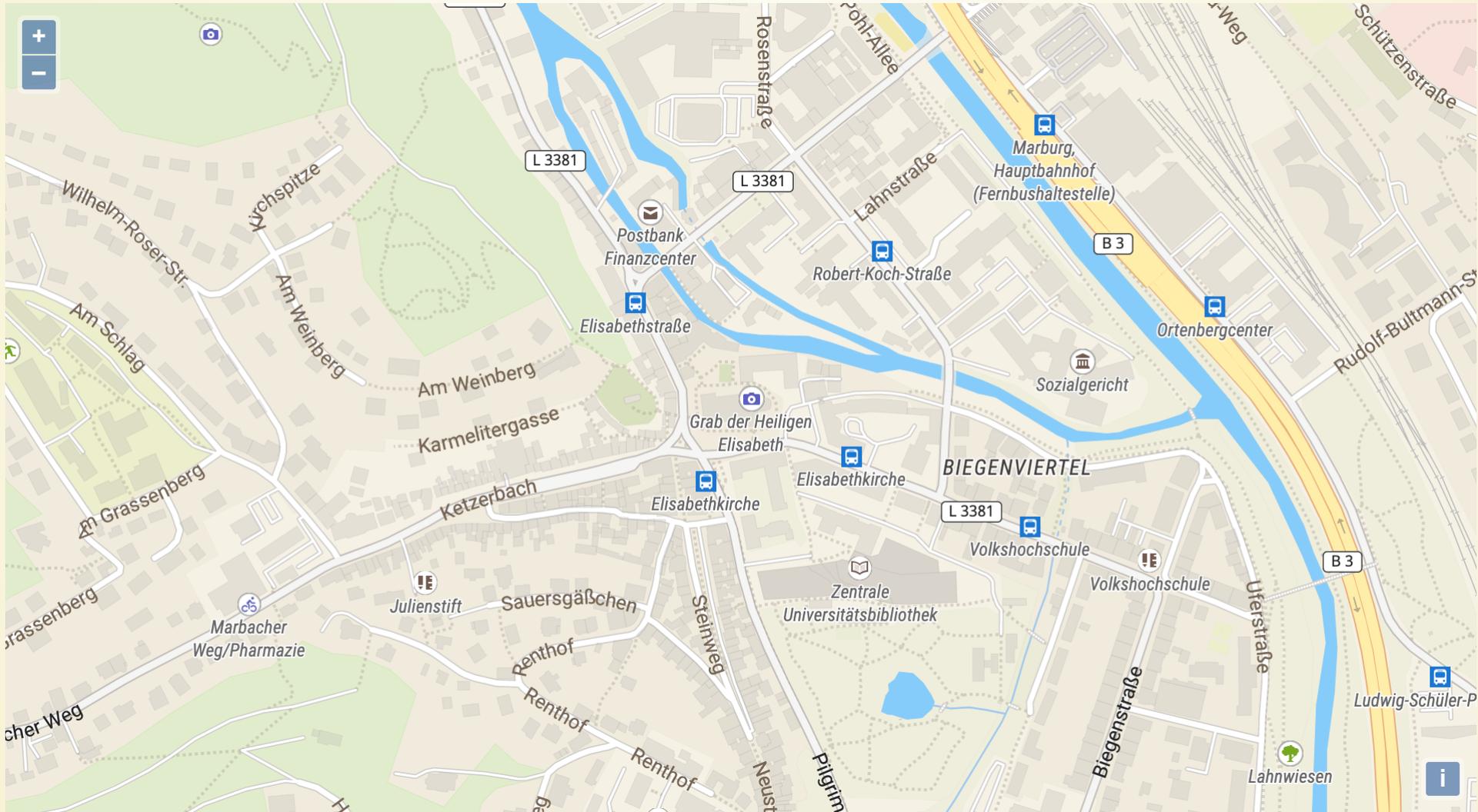


Vector Tiles

Mapbox Vector Tiles + Mapbox Style
(Mapbox, Maptiler, Geoapify, Carto, Esri, ...)

```
import MapboxVectorLayer from 'ol/layer/MapboxVector.js';  
  
const layer = new MapboxVectorLayer({  
  styleUrl: 'https://api.maptiler.com/maps/streets/style.json?key  
});
```

Vector Tiles



Simplify & Modify

Beispiele für Performance-Optimierungen

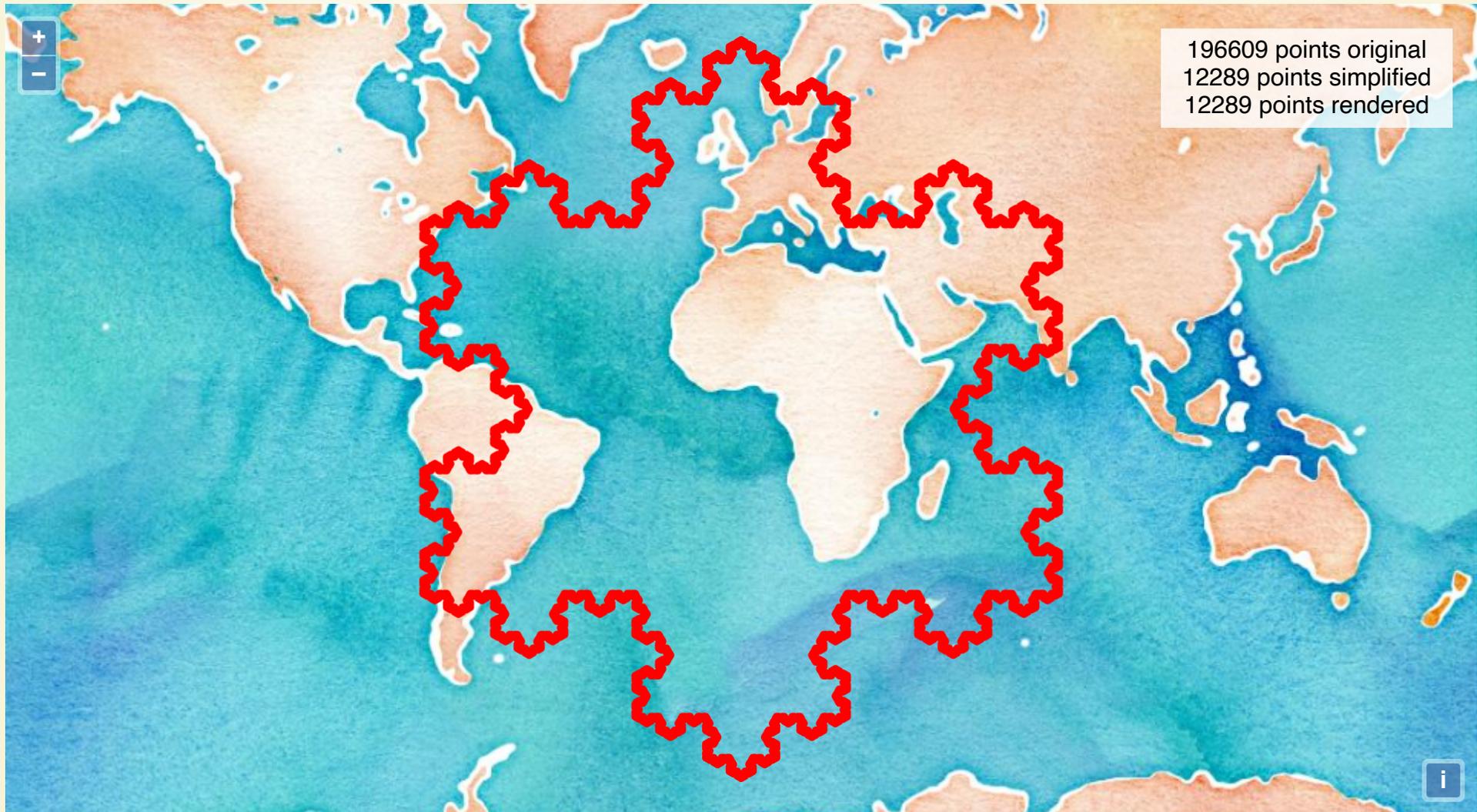
- Vektordaten-Reduktion beim Rendern (Douglas-Peucker, Quantizing)
- Modify Interaction arbeitet segmentbasiert mit RTree von Vertices

Simplify & Modify

Und was habe ich davon?

- Gute Render-Performance auch ohne WebGL
- Bearbeiten von Geometrien mit 10x mehr Vertices als mit mapbox-gl-draw

Simplify & Modify



Lokale Projektionen

OpenLayers kann Karten beliebig umprojizieren.

Auch Rasterlayer 

Aber brauche ich das?

Wahrscheinlich nicht!

Lokale Projektionen

```
import proj4 from "proj4";
import { register } from "ol/proj/proj4.js";
import { setUserProjection } from "ol/proj.js";

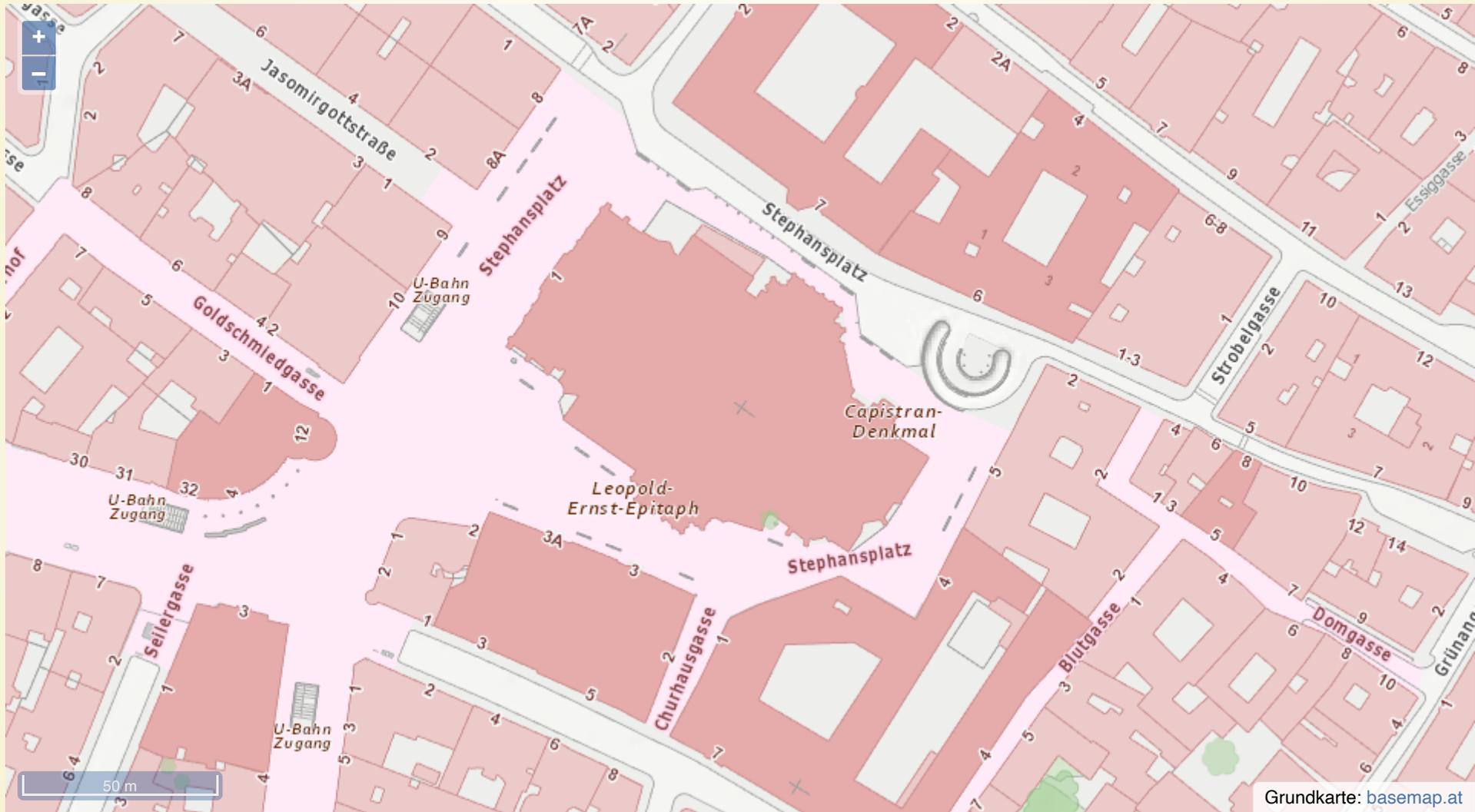
proj4.defs('EPSG:31259', '+proj=tmerc +lat_0=0 +lon_0=16.333333333333334');
register(proj4);

setUserProjection('EPSG:31259');
```

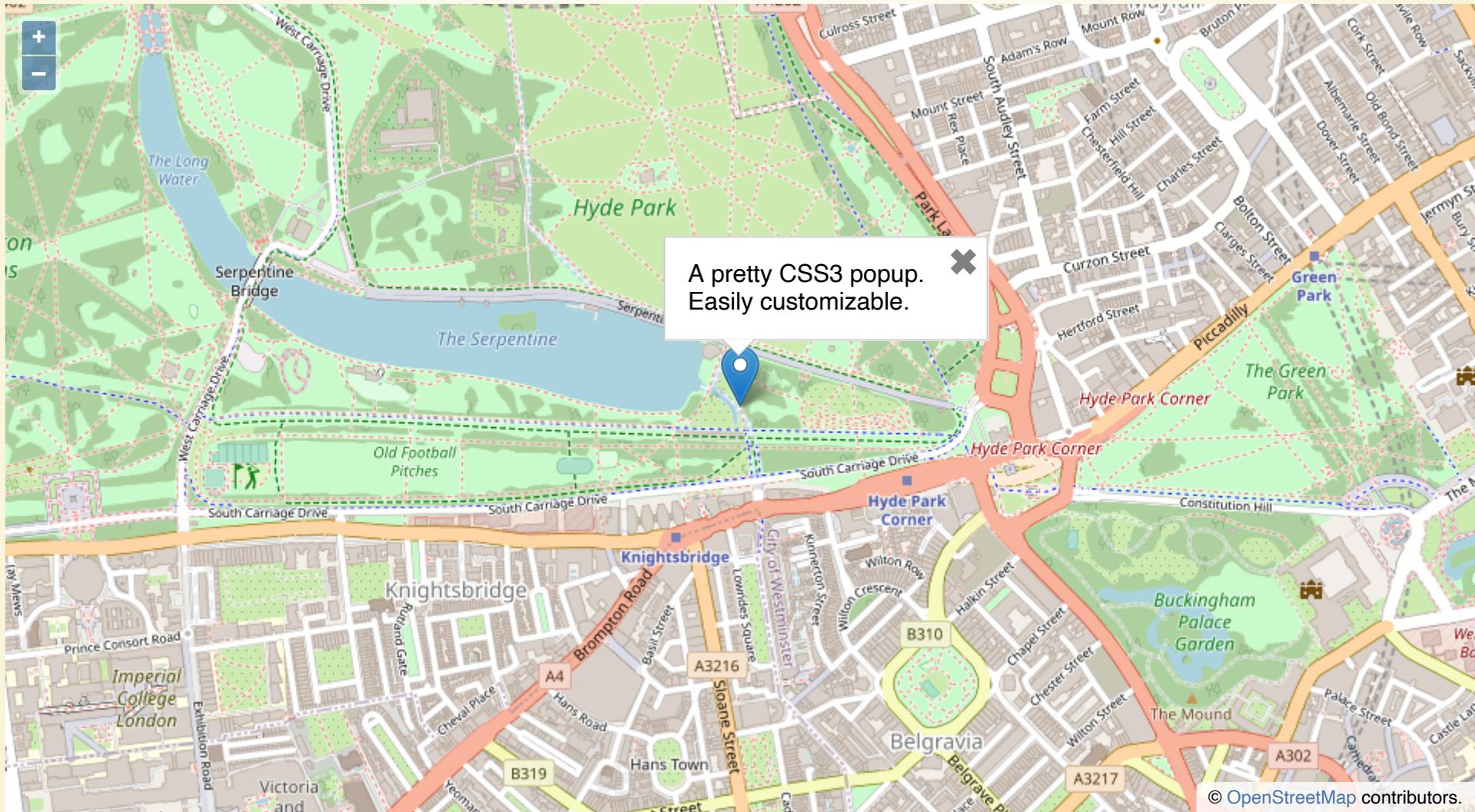
Von nun an ist alles in **MGI / Austria GK M34** 🎉

Nur die Karte bleibt in Web Mercator ✅

Lokale Projektionen



Leaflet ist einfacher



Leaflet ist einfacher

```
const map = L.map('map').setView([51.505, -0.09], 13);
```

```
useGeographic();
```

```
const map = new Map({  
  target: 'map',  
  view: new View({ center: [-0.09, 51.505], zoom: 13 })  
});
```

Leaflet ist einfacher

```
L.tileLayer('https://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png',  
  attribution: '© <a href="https://www.openstreetmap.org/copyrigh  
  }).addTo(map);
```

```
map.addLayer(new TileLayer({ source: new OSM() }));
```

Leaflet ist einfacher

```
const marker = L.marker([51.5, -0.09]);  
marker.addTo(map);
```

```
const marker = new Marker([-0.09, 51.5]);  
marker.setMap(map);
```

Marker: Andreas Hocevars 3d Party Library
[ol-marker-feature](#)

Leaflet ist einfacher

```
marker
  .bindPopup('A pretty CSS3 popup.<br> Easily customizable.')
  .openPopup();
```

```
const popup = new Popup({ offset: [0, -32] });
map.addOverlay(popup);

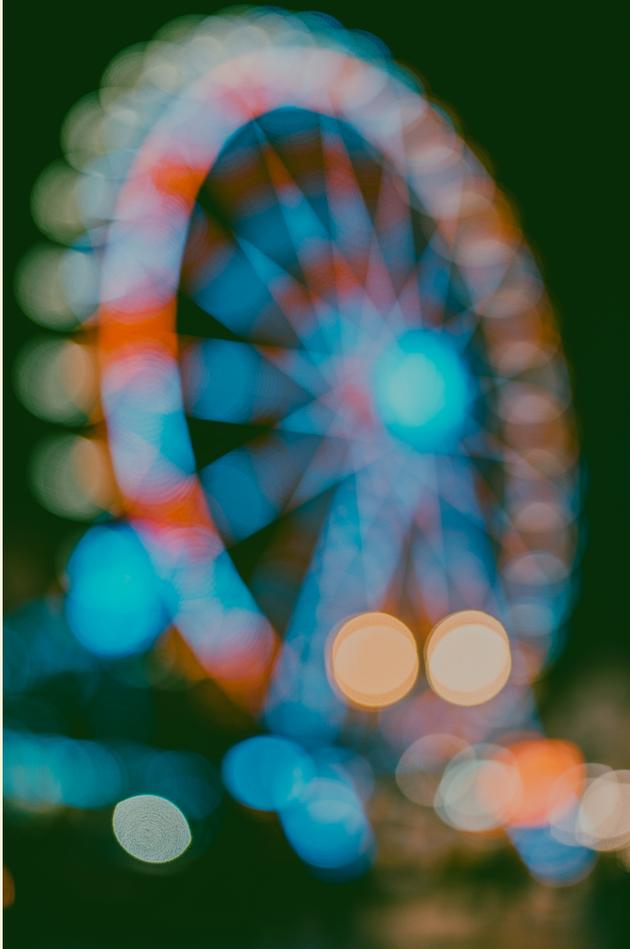
marker.on('click', () => popup.show(marker.getLocation(),
  'A pretty CSS3 popup.<br> Easily customizable.));
marker.dispatchEvent('click');
```

Popup: Matt Walkers 3d Party Library

ol-popup

Ausblick 👁️👁️

Zukünftige Features



- **Style**
Rich Text Labels
- **Performance**
Mehr Web Worker, mehr WebGL
- **Standards**
Aspekte von OGC APIs, STAC + Extensions
- **Typings**
Strict Null Checks, Union Types statt enums
- **+ Eure Beiträge** 🙌
- ...

Vielen Dank

**Fragen &
Anmerkungen?**

Impressum

Impressum

Autoren

Marc Jansen

terrestris GmbH & Co. KG

Kölnstraße 99

53111 Bonn, Germany

jansen@terrestris.de

Andreas Hocevar

ahocevar geospatial

Grüne Gasse 21d/25,

8020 Graz, Austria

mail@ahocevar.com

Lizenz

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

Vortragsfolien, PDF-Version, git repository