



2700 interaktive thematische Karten - Ein Fall für Vector Tiles!

Pirmin Kalberer
@implgeo
Sourcepole AG, Zürich
www.sourcepole.ch



Schweizer Brutvogelatlas 2013–2016





Feldlerche

Alauda arvensis
Alouette des champs
Allodola
Iodola da prada
Eurasian Skylark



Rote Liste potenziell gefährdet (NT)
Bestand 25000–30000 Paare (2013–2016)

Focus Seiten 362, 402, 414, 428, 478, 494, 506

¹Aalkoet et al. (2016); ²Beaud & Beaud (2018); ³Berger-Flückiger et al. (2008b); ⁴Bundesamt für Raumentwicklung (2014); ⁵Fischer et al. (2009); ⁶Glutz von Blotzhorn (2007a); ⁷Graf et al. (2014b); ⁸Jenny et al. (2014a–b); ⁹Körner et al. (2017); ¹⁰Martinez & Berner (2017); ¹¹Müller & Erni (2014); ¹²Müller & Wiggler (2018); ¹³Schläpfer (1988); ¹⁴Schmidt et al. (2017); ¹⁵Siero et al. (2009)

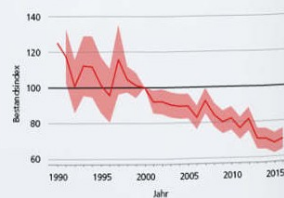
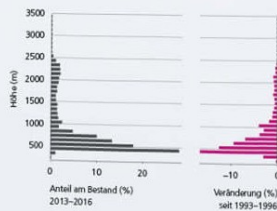
In der Schweiz brütet die Feldlerche heute vor allem in trockenen Getreideanbaugebieten im westlichen Mittelland, in der Ajoie JU und im Klettgau SH zwischen 400 und 800 m. Sie besiedelt auch alpine Gebiete. Einige Hochplateaus auf rund 2000 m weisen erstaunlich hohe Dichten auf, so der Schamserberg GR mit lokal bis 5 Revieren/10 ha¹. Die höchsten Brutnachweise gelangen bei Evolène VS auf 2460 m (M. Thélin) und bei Albinen VS auf 2420 m (S. Stöckli). Die höchsten Sänger wurden früher bei Zinal VS auf 2860 m¹⁰ und aktuell bei Zermatt VS auf 2790 m (C. Hühler) entdeckt. Als ursprüngliche Steppenart besiedelt die Feldlerche offene, weite Landschaften und hält Abstand zu Vertikalstrukturen (Wälder, Siedlungen und Freileitungen). Sie bevorzugt relativ niedrige und lückige Vegetation und brütet in Tallagen fast ausschliesslich in Ackerkulturen. Die intensive Grünlandbewirtschaftung mit kurzen Mahdintervallen verunmöglicht ein erfolgreiches Brutgeschäft zunehmend auch in höheren Lagen^{2, 9}. In alpinen Gebieten bevorzugt die Feldlerche weite Mähwiesen und Kuppenlagen.

Seit 1993–1996 ist das Areal weiter geschrumpft und die Dichte hat fast überall abgenommen¹⁵. Im Mittelland ist die Dichte mittlerweile wohl rund zehnmal

tiefer als um 1990^{13, 15}. Die Bestände sind auf allen Höhenstufen zurückgegangen. Regionale Untersuchungen ergaben Rückgänge seit etwa 1990 von 50 bis 77 %^{10, 11}. Im Kanton Zürich sank der Bestand von 2008 bis 2017 um 54 %¹². Auch in höheren Lagen nahm er ab, so im Engadin GR zwischen 1988 und 2010 um 44 %⁹.

Die Hauptursache für den Negativtrend, der in ganz Europa (über 50 % Rückgang seit 1980) verzeichnet wird^{16, 17}, ist eine weitere Intensivierung der Landwirtschaft, einerseits durch die Landwirtschaft, aber zunehmend auch durch die Zersiedelung der Landschaft⁴. Um die Feldlerche zu erhalten, sollen deshalb vermehrt auch raumplanerische Aspekte berücksichtigt werden. Verbliebene offene Kulturlandschaften müssen vor weiterer Überbauung und Fragmentierung geschützt werden. Gleichzeitig müssen dringend grossräumig Fördermassnahmen in Ackerkulturen (z.B. Bunt- und Rotationsbrachen, weitgesätetes Getreide, Untersaaten usw.) umgesetzt werden^{5, 8, 14}. Brutten in Wiesengebieten sollen mit Spätmahdverträgen geschützt werden^{7, 8}.

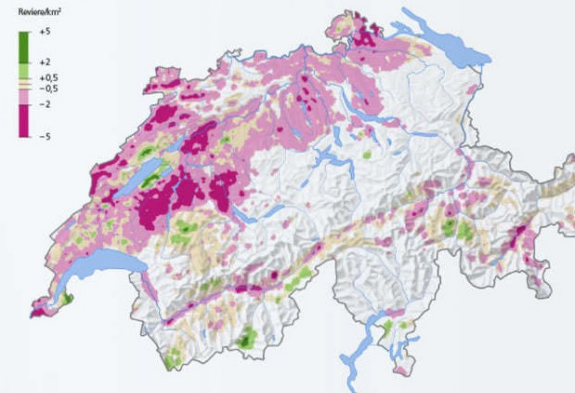
Judith Zellweger-Fischer



Dichte 2013–2016



Veränderung der Dichte seit 1993–1996



Patenschaft
Alex Schläpfer
Urs-Peter Stäubli





Technische Randbedingungen

- › 420 Vogelarten
- › 2700 thematische Karten
- › 2D und 3D Karten
- › Interaktiv (Mouseover, Timesliders, etc.)
- › Darstellung in LV95
- › Einbettung in CMS
- › Responsive Layout
- › 4 Sprachen



The screenshot shows the website interface for 'vogelwarte.ch'. The main navigation bar includes 'Atlas', 'Kontakt', 'News & Medien', 'Sprache', and a search bar. Below the navigation, there are buttons for 'Vogelwarte', 'Besuch', 'Projekte', 'Vögel', and 'Shop'. The page title is 'Vögel', and the breadcrumb trail is 'Vögel > Vögel der Schweiz > Adlerbussard > Karten und Diagramme'. A grid of nine map thumbnails is displayed, with 'Veränderung der Dichte' selected. The main map area shows a topographic map of Switzerland with a density change overlay for 'Bachstelze' from 1993-96 to 2013-16. A legend on the left indicates a color scale from -3 to 3 'Revier/km²', with a text box explaining that the map shows the change in the number of territories per km². A 'Jahreskalender' on the right shows the bird's status by month. The bottom left has 'Relief' and 'Gemeinden' checkboxes, and the bottom right has a 'Vergleich' button.



Umsetzung

- › **Import der Geodaten in PostGIS**
- › **Metadaten in PostGIS**
- › **Vector-Tile Generierung mit t-rex**
- › **Styling mit Mapbox GL JSON**
- › **Kartenapplikation mit Mapbox GL JS**
- › **Generierung Legenden, Texte, etc. aus Metadaten**



Datenaufbereitung

- › **Vektordaten in GPKG-Format**
 - › Import mit ogr2ogr
- › **Rasterdaten in GeoTIFF-Format**
 - › Vektorisierung mit Python / GDAL



Vector-Tile Generierung

- › Mapbox Vector Tiles
- › Datenquelle PostGIS
- › t-rex
 - › <https://t-rex.tileservers.ch/>



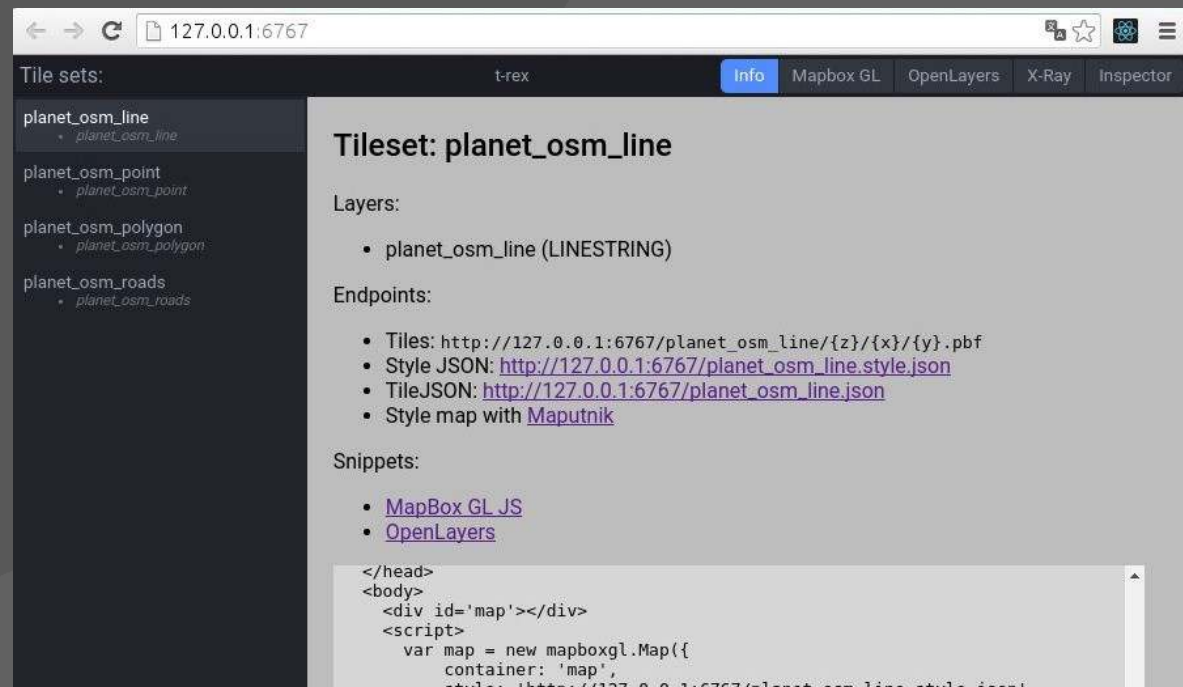
Serve vector tiles

- ★ Live tiles from PostGIS geodata
- ★ Zero-configuration mode



› Serve vector tiles

- › Live tiles from PostGIS databases and GDAL vector formats
- › Zero-configuration mode
- › Embedded webserver
- › Visual styling with Maputnik



Tile sets:

- planet_osm_line
 - planet_osm_line
- planet_osm_point
 - planet_osm_point
- planet_osm_polygon
 - planet_osm_polygon
- planet_osm_roads
 - planet_osm_roads

Tileset: planet_osm_line

Layers:

- planet_osm_line (LINESTRING)

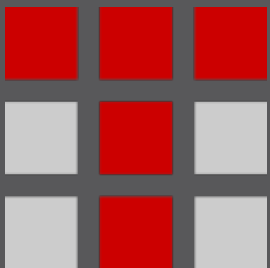
Endpoints:

- Tiles: http://127.0.0.1:6767/planet_osm_line/{z}/{x}/{y}.pbf
- Style JSON: http://127.0.0.1:6767/planet_osm_line.style.json
- TileJSON: http://127.0.0.1:6767/planet_osm_line.json
- Style map with [Maputnik](#)

Snippets:

- [MapBox GL JS](#)
- [OpenLayers](#)

```
</head>
<body>
  <div id='map'></div>
  <script>
    var map = new mapboxgl.Map({
      container: 'map',
      style: 'http://127.0.0.1:6767/planet_osm_line.style.json'
```



› Generate vector tiles

- › Tile generation command with simple parallelization
- › Generate configuration template
- › Support for custom tile grids

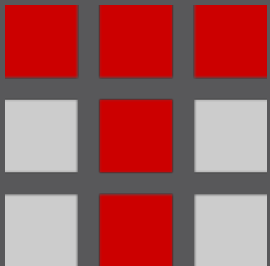
```
pi ~ t_rex generate -h
t_rex-generate
Generate tiles for cache

USAGE:
  t_rex generate [OPTIONS] --config <FILE>

FLAGS:
  -h, --help      Prints help information
  -V, --version   Prints version information

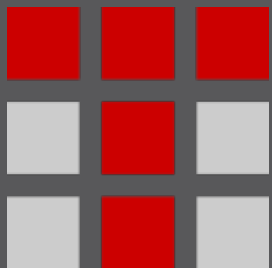
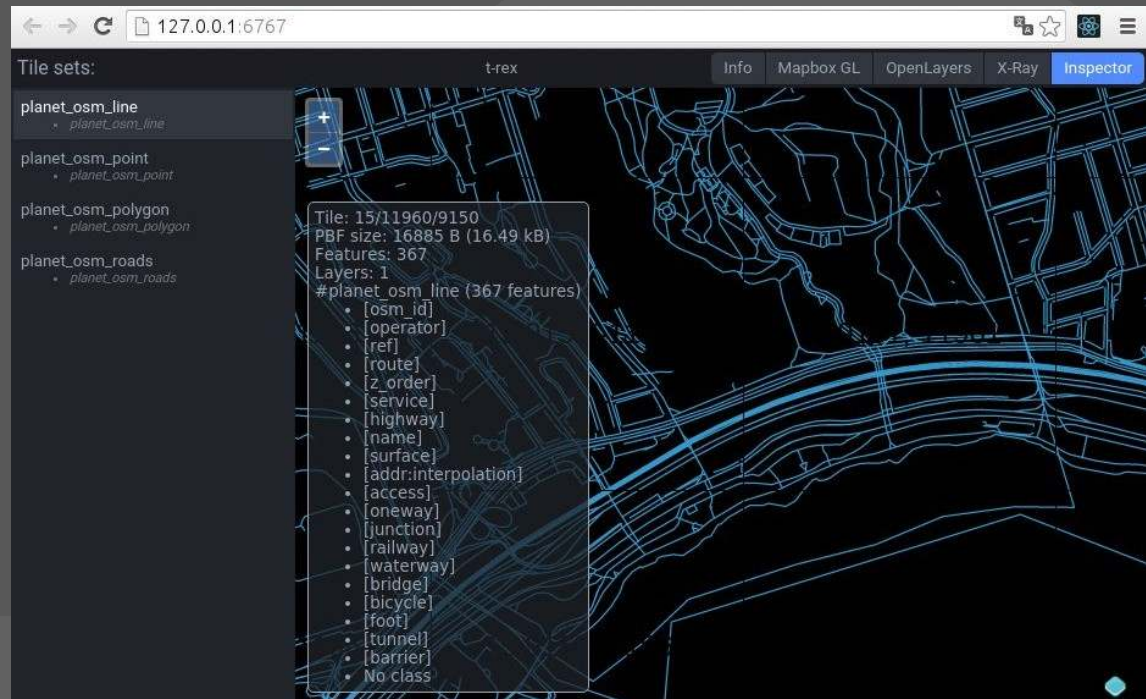
OPTIONS:
  -c, --config <FILE>          Load from custom config file
  --extent <minx,miny,maxx,maxy> Extent of tiles
  --maxzoom <LEVEL>           Maximum zoom level
  --minzoom <LEVEL>          Minimum zoom level
  --nodeno <NUM>              Number of this nodes (0 <= n < nodes)
  --nodes <NUM>              Number of generator nodes
  --progress <true|false>    Show progress bar
  --tileset <NAME>           Tileset name
```

```
pi ~ t_rex generate --config natural_earth_vectors.cfg --minzoom=3 --maxzoom=5
2016-09-12 11:33:45.719 INFO Reading configuration from 'natural_earth_vectors.cfg'
```



➤ Easy to Use

- Auto-detection of layers in database
- Single human readable configuration file
- Automatic reprojection to grid CRS
- Install package or run in a Docker container





Demosession

- › **Vektor-Tiles Karten erstellen mit t-rex**
- › **Freitag 11:00 HS Anatomie**



Aufteilung in Tilesets

› Variante 1:

- › 1 Vogelart * 1 Layer pro Kartentyp * XYZ

› Variante 2:

- › 1 Kartentyp * 1 Layer pro Vogelart * XYZ

› Kriterien:

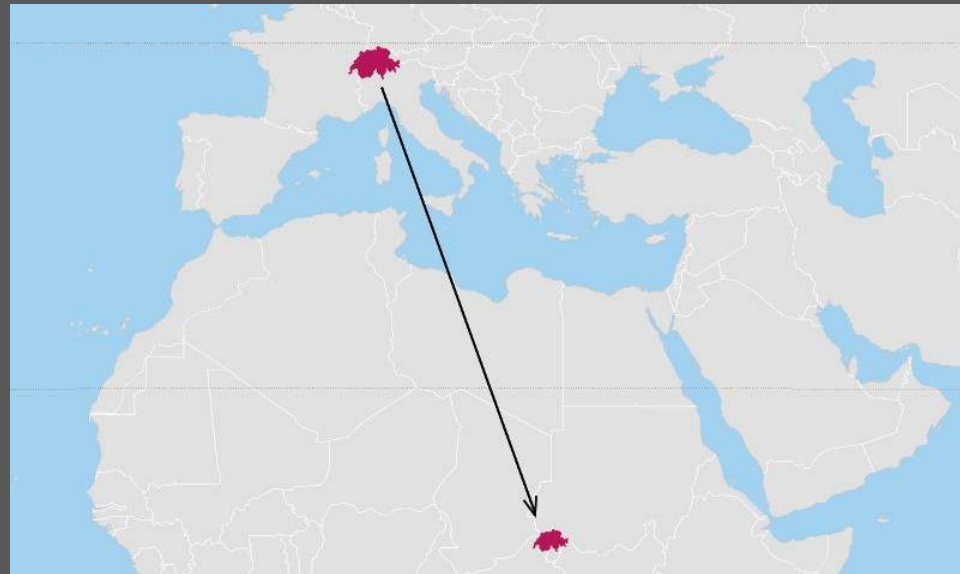
- › Kachelgrösse
- › Userverhalten (Mehr Arten- oder Kartenwechsel?)

› Entscheid: Variante 1



Relief / LV95

- Mapbox GL JS: Web Mercator only
- Serving Non-mercator data in a Web Mercator grid



- Crop Relief
 - `gdalwarp -cutline ch.shp -crop_to_cutline ...`
- Fake Mercator extent of relief:
 - `t_rex genconfig --datasource ch.shp --no-transform true | grep extent`



Tilesets für 420 Arten

↳ t-rex Templating Funktion

```
[grid]
```

```
predefined = "web_mercator"
```

```
{% for id in species_ids %}
```

```
[[tileset]]
```

```
name = "{{id}}"
```

```
* Real extent: [5.97203, 45.81704, 10.53440, 47.77335]
```

```
extent = [22.33212, 9.61149, 25.44029, 11.56287]
```

```
minzoom = 6
```

```
maxzoom = 10
```

```
[[tileset.layer]]
```

```
name = "density"
```

```
fid_field = "ogc_fid"
```

```
geometry_field = "wkb_geometry"
```

```
geometry_type = "POLYGON"
```

```
srid = 2056
```

```
no_transform = true
```

```
buffer_size = 1
```

```
simplify = true
```

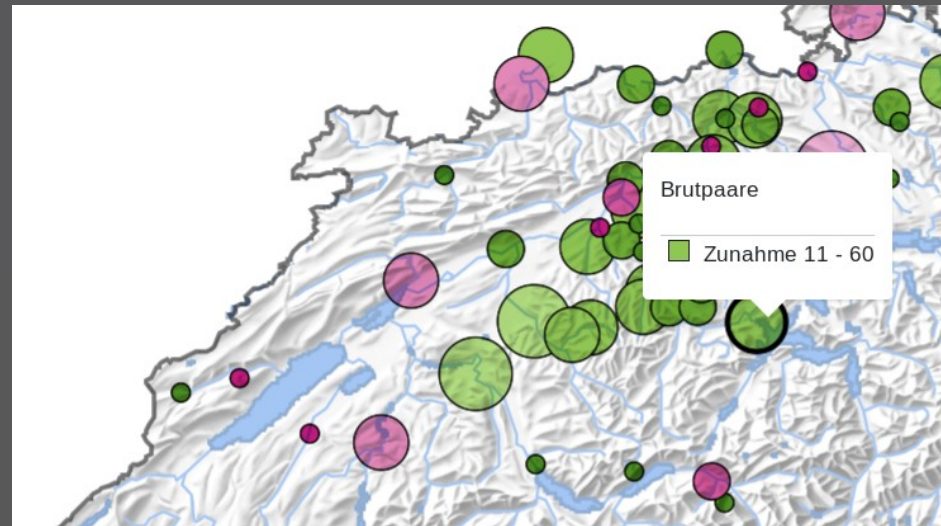
```
[[tileset.layer.query]]
```

```
sql = ""SELECT wkb_geometry,ogc_fid,value FROM birddata.density WHERE  
species_id={{id}} AND wkb_geometry && !bbox!""
```



Mapbox GL JSON Styles

```
{  
  "source": "birddata",  
  "source-layer": "change_increase",  
  "type": "circle",  
  "paint": {  
    "circle-radius": {  
      "property": "code",  
      "stops": [  
        [1, 5],  
        [2, 10],  
        [3, 15],  
        [4, 20]  
      ]  
    },  
    "circle-stroke-width": ["case",  
      ["boolean", ["feature-state", "hover"], false],  
      3,  
      1  
    ],  
    "circle-stroke-color": "#000000"  
  }  
},
```



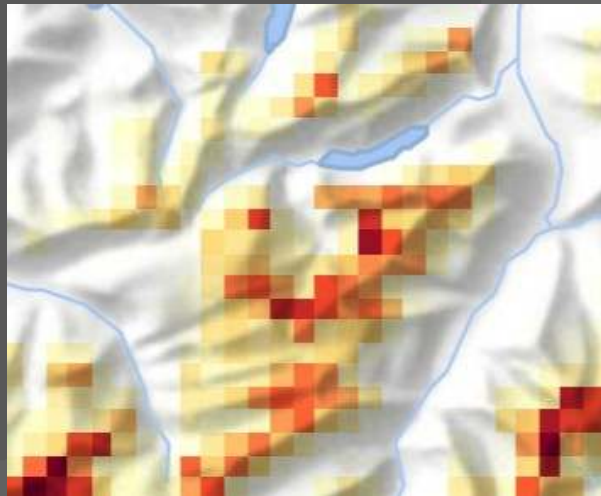


Blending Modes

Alpha-Transparenz:



Blending Mode Color Multiply:





Fazit

- › **Vektor Tiles eignen sich sehr gut für thematische Overlays**
- › **Gute Performance für mobile Viewer**
- › **Interaktivität wesentlich einfacher realisierbar**
- › **Betrieb ohne Kartenserver**



Danke!



Pirmin Kalberer
@implgeo