



FOSS4G 2024

Serving live maps with vector tiles

Pirmin Kalberer
@implgeo
Sourcepole, Zurich
www.sourcepole.com



SOURCEPOLE
Linux & Open Source Solutions





Use cases

- › **Maps with sensor values**
 - › Weather
 - › IoT applications (traffic counters, ...)
- › **Object positions**
 - › Fleet tracking
 - › Air traffic
 - › Public transport (trains, etc.)



Live data formats in use

› GeoJSON

- › Simple and flexible
- › Inefficient
- › Performance workaround: Display clusters

› Raster Tiles

- › Inefficient for updates
- › No client-side info request
 - › Workaround: UTFGrid





Vector Tiles

- › **Small size (mobile usage possible)**
- › **Client-Side rendering and quering**
 - › Flexible Layer de-/activation
- › **Cachable**
- › **Disadvantages:**
 - › Tiling artifacts (splitting of areas, feature duplication, labelling)





Vector tile storage formats (static)

› Single file per tile

- › + Good for updating
- › + Support from CDN's (S3 protocol)
- › - Possibly millions of files

› MBTiles

- › + Single file
- › + Good for updating
- › - Requires tile service for delivery

› PMTiles

- › + Single file
- › + Compact
- › + Only HTTP server required (Range Requests)
- › - Not well suited for partial updates





Tile servers (live data from PostGIS)

- › **Universal map servers**
 - › GeoServer
 - › MapServer
- › **Specialized**
 - › BBOX (T-Rex successor)
 - › Martin
 - › many more...





Publishing real-time updates

- › **Javascript Update Loop (Polling)**
 - › Custom optimizations like delivery of changed data only
- › **WebSockets**
 - › or Long Polling, Server Sent Events (SSE), etc.
 - › + Efficient
 - › + Real-time updates
 - › - Code/Style duplication
 - › - Scaling is difficult
- › **Update based on Caching Headers**
 - › MapLibre: built-in
- › **Additional optimizations**
 - › Display Interpolated values





Demo 1

› MOTIS

- › Intermodal Mobility Information System
- › <https://github.com/motis-project/motis>
- › Demo: Transitous
 - › <https://transitous.org/>
- › Map Tiles from built-in vector tile server
- › Traffic data as JSON (Bounding box requests with polling)





Demo 2

› BBOX server live demo

- › www.bbox.earth
- › Live data Overlay (demo, no serve-side caching)
- › Vector tiles for live data
- › Zoom-level dependent cache headers





Tile server configuration

› BBOX layer query

```
[[tileset.postgis.layer.query]]
sql = """
    SELECT *
    FROM last_positions
    WHERE ts > NOW() - INTERVAL '2 hours'
    """
```

› BBOX cache configuration

```
[[tileset.cache_control]]
maxzoom = 7
max_age = 60
```

```
[[tileset.cache_control]]
minzoom = 8
max_age = 5
```





Client configuration

› MapLibre source

```
"sources": {  
  "realtime": {  
    "type": "vector",  
    "url": "https://tiles.bbox.earth/xyz/realtime.json"  
  }  
},
```





› MapLibre style

```
{
  "id": "position-icons",
  "type": "symbol",
  "source": "realtime",
  "source-layer": "positions",
  "filter": [ "in", [ "get", "symbol_code" ], "&" ],
  "layout": {
    "symbol-placement": "point",
    "icon-allow-overlap": true,
    "icon-ignore-placement": true,
    "icon-size": { "stops": [ [ 6, 0.8 ], [ 12, 1 ], [ 14, 1.5 ] ] },
    "icon-image": [ "match", [ "get", "symbol_code" ],
      "&", "icon-viewpoint",
      "transport-information"
    ]
  },
  "paint": {
    "icon-color": "#66626a"
  },
}
```



Wrapup

- › **Vector tiles are an efficient way to serve live maps**
- › **No workarounds like clustering needed**
- › **Optimize tile size (style ↔ data content)**
- › **Use caching headers**
 - › **CDN support**
- › **Side note: Please don't use Web-Mercator for world maps**





Thank you!



Pirmin Kalberer
@implgeo