



MapComponents

Open Source 3D-GIS im Browser: Einstieg in 3D Tiles mit deck.gl und MapComponents

26.03.2026 | Max Tobias Weber



Max Tobias Weber

- ▶ 2007 - 2021 Entwickler bei dahabtours
- ▶ 2016 Mitgründung der dahabtours GmbH
- ▶ 2010 - 2020 Projekte Backend & Frontend für MeineStadt, AOK, Mujumaps, PopNRW, Tierversicherung.biz, Goerlitz AG, ...
- ▶ seit 2021 als Entwickler bei der WhereGroup GmbH in Bonn
- ▶ MapComponents, Mapbender, QGIS, Javascript, React, MapLibre, Docker, Kubernetes ...

tobias.weber@wherogroup.com





Workshop

- ▶ Einleitung: Überblick über MapComponents & MapComponents deck.gl Package
- ▶ Praxisteil: Entwicklung einer einfachen Web-Kartenanwendung mit MapComponents & 3D Tiles



Was sind MapComponents?

MapComponents ist ein clientseitiges React Framework zur deklarativen Erstellung interaktiver Kartenanwendungen mithilfe von selbstorganisierenden, isolierten Komponenten.



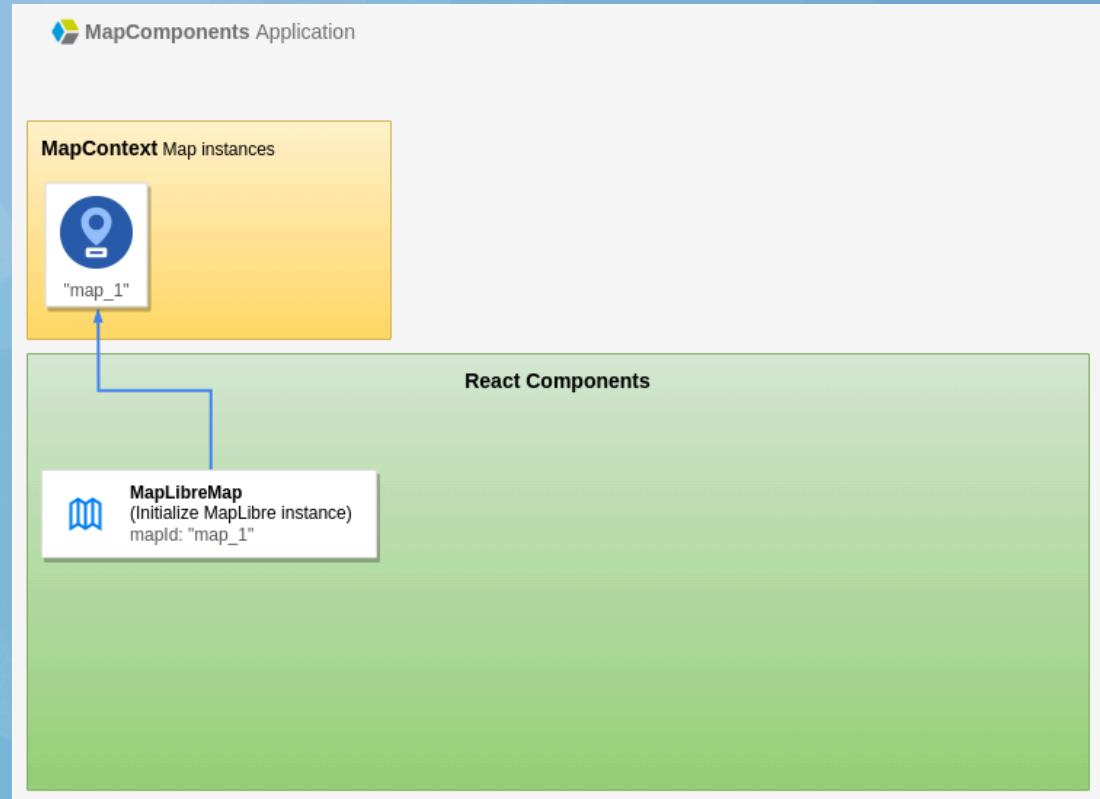
MapContext

- ▶ Zentrale Registrierung für MapLibre-gl Instanzen
- ▶ Stellt Funktionen zur Registrierung und zum Abruf von MapLibre-gl Instanzen bereit



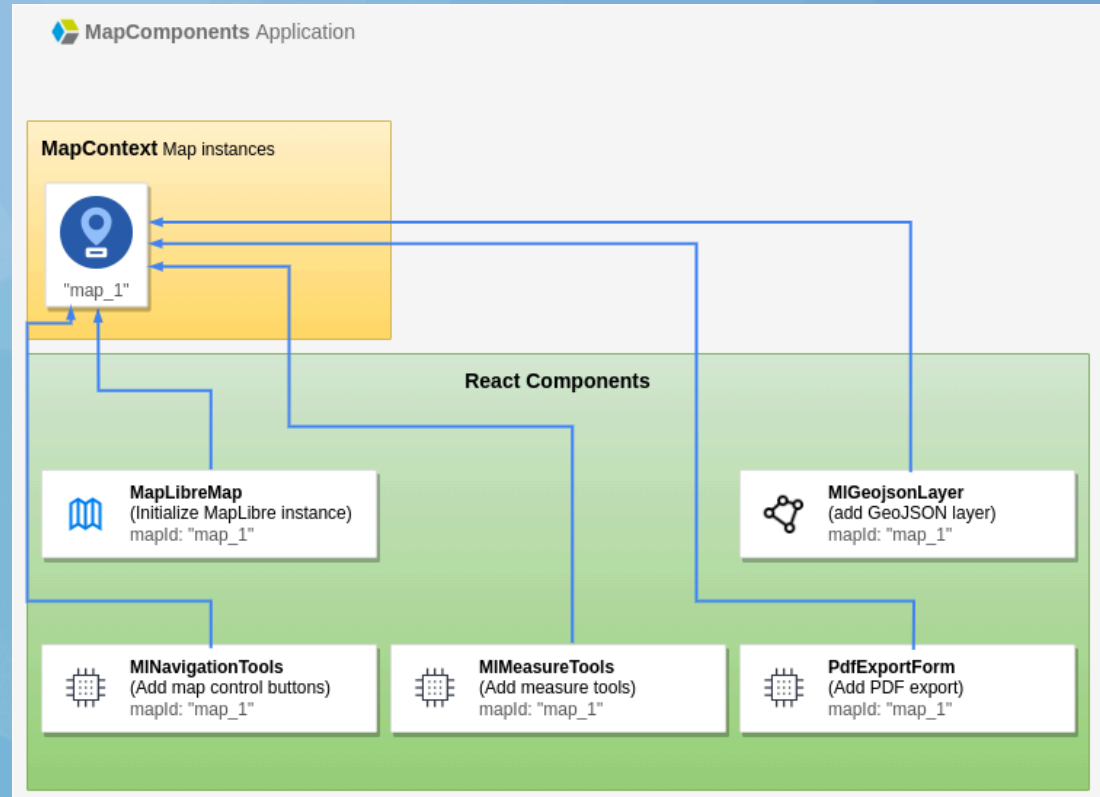
MapLibreMap

- ▶ Initialisiert eine MapLibre-gl Instanz
- ▶ Registriert sie im MapContext



Eigenschaften eines MapComponents

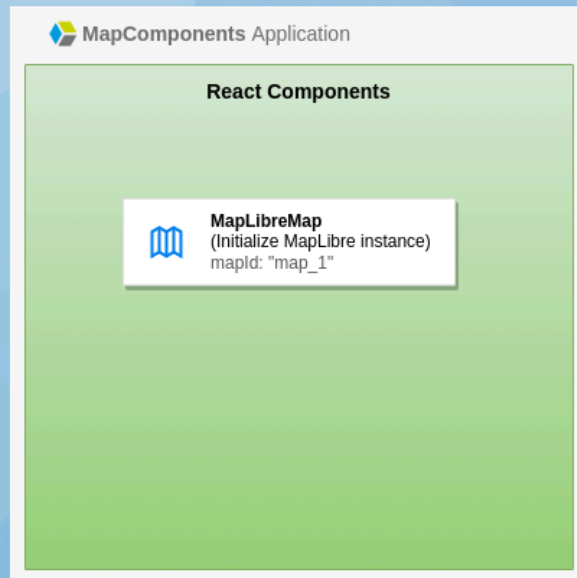
- ▶ Abruf einer MapLibre Instanz aus dem MapContext
- ▶ Überwachen des Zustands
- ▶ Manipulieren des Zustands
- ▶ Reagieren auf Zustandsänderungen
- ▶ Alle nicht persistenten Zustandsänderungen aufräumen





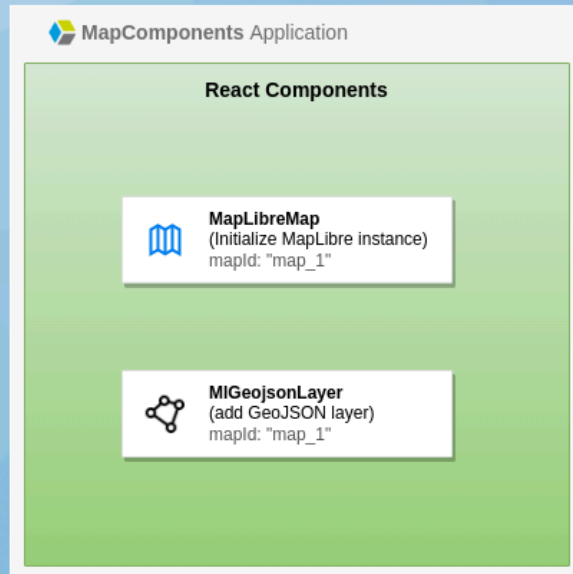
Beispiel für eine einfache Kartenanwendung

- ▶ Ein MapComponentsProvider und ein MapLibreMap Component



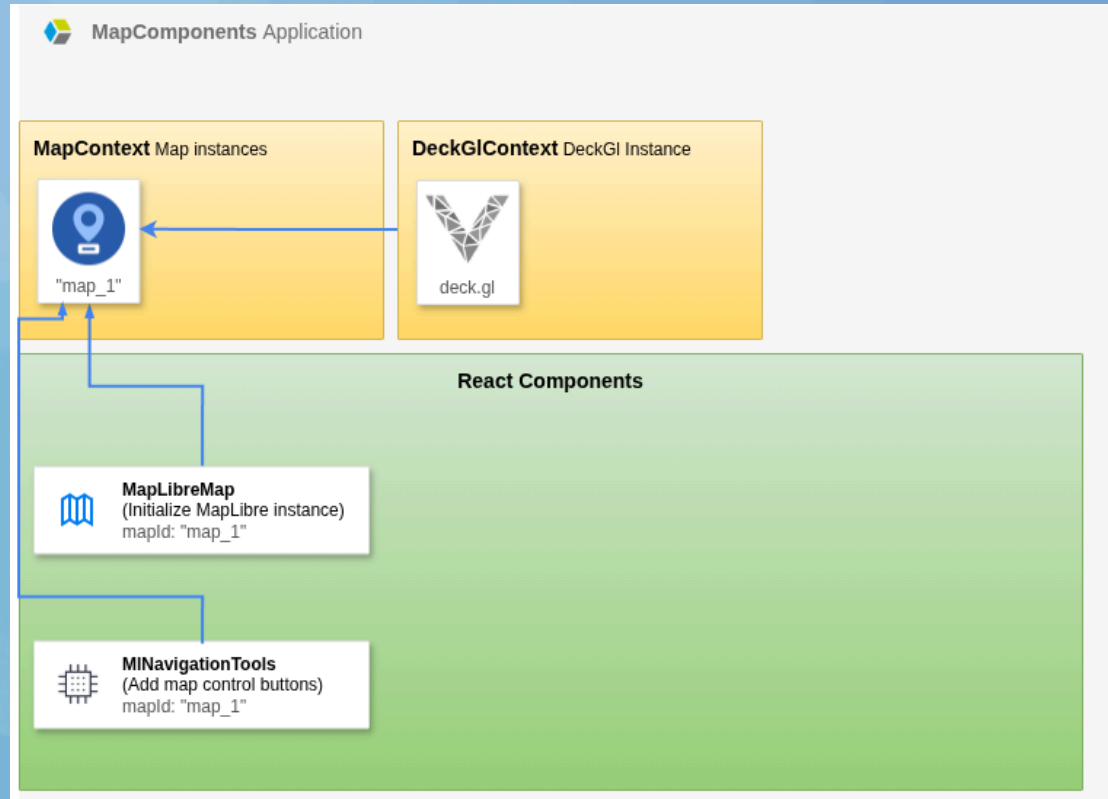
Beispiel: Darstellung von Geometrien

- ▶ Ein MGeoJsonLayer zur Darstellung eines Polygon GeoJSONs
- ▶ Wird das MGeoJsonLayer Component wieder aus dem React-DOM entfernt, entfernt es auch alle Sources, Layer und Event Handler, die es zur MapLibre-gl Instanz hinzugefügt hat



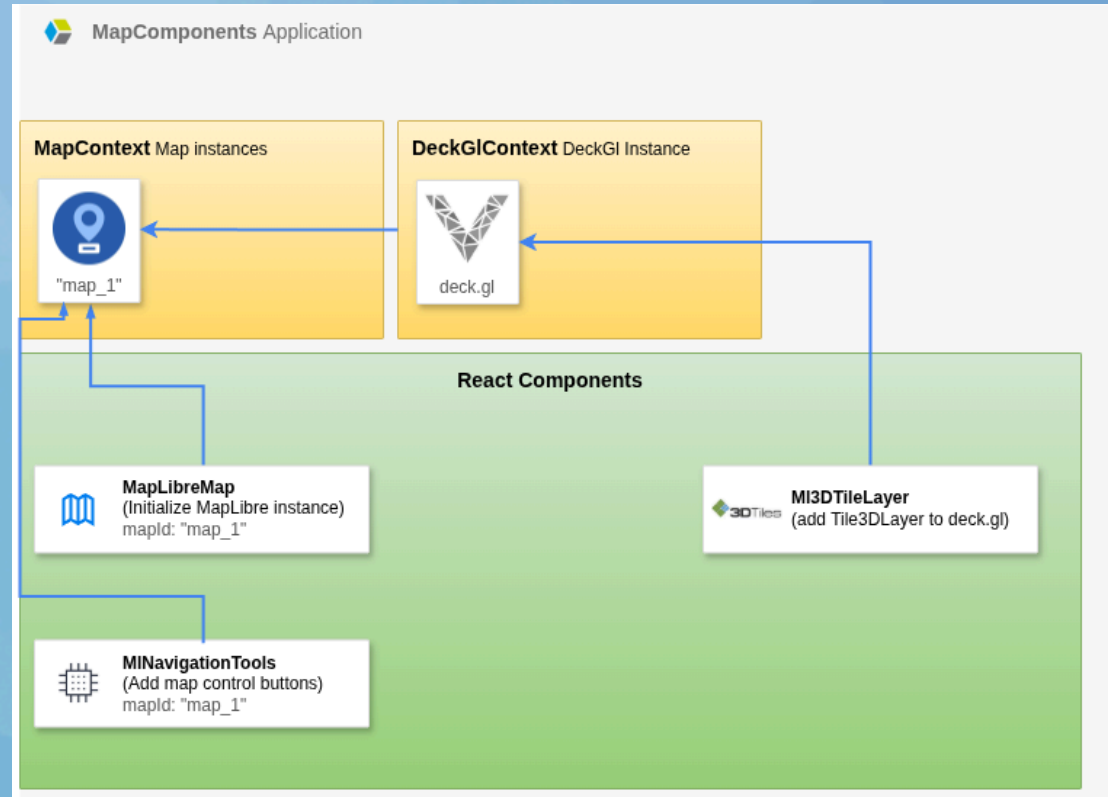
Deck.gl Integration

- ▶ DeckGlContextProvider erzeugt eine deck.gl Instanz
- ▶ Erzeugt MapboxOverlay in der MapLibre Instanz aus dem MapContext
- ▶ Ermöglicht das hinzufügen und entfernen von deck.gl Layern



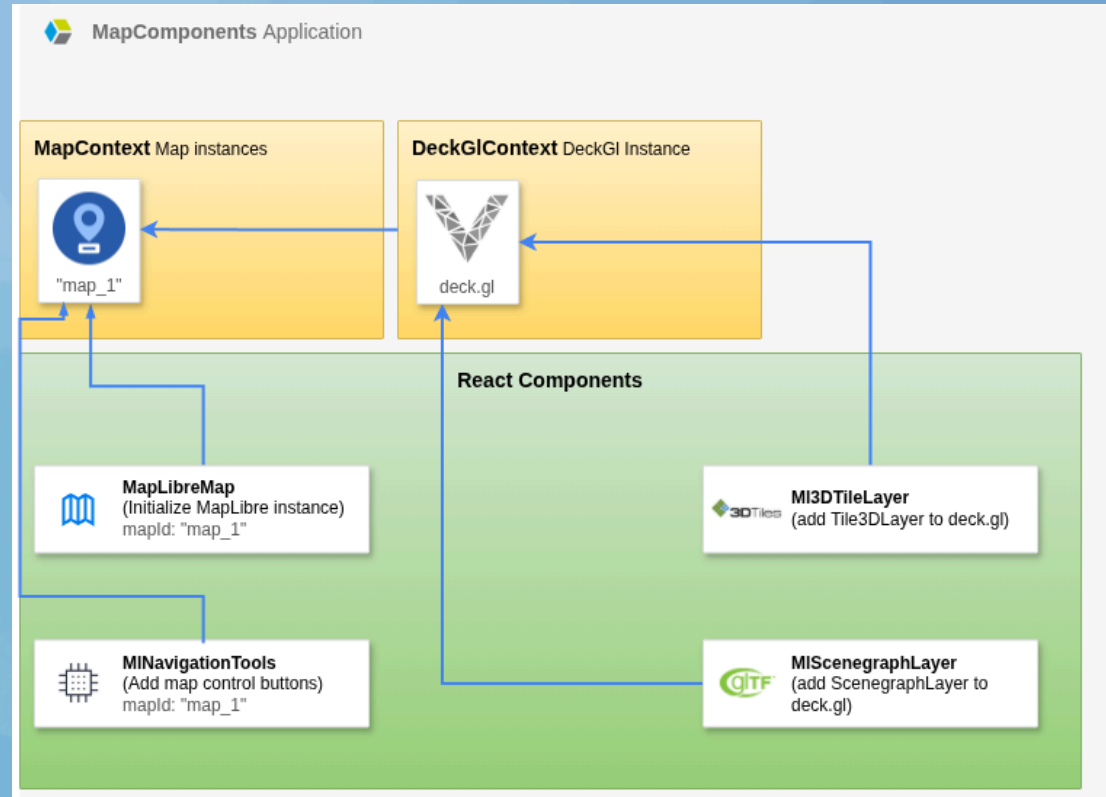
MI3DTileLayer

- ▶ Darstellung von 3D Tiles Daten mit dem Tile3DLayer von deck.gl
- ▶ Rendert 3D Tiles Daten als PointCloudLayer, a ScenegraphLayer or SimpleMeshLayer
- ▶ Leider momentan noch schlechte Unterstützung für angepasste Darstellung und Interaktion



*MI*ScenegraphLayer

- ▶ Basierend auf dem ScenegraphLayer von deck.gl
- ▶ Ermöglicht die effiziente Darstellung von Punktgeometrien als 3D Modelle





MapComponents vite Template

"Bootstrap a mapcomponents app by running a single command."

Basis Projektsetup auf der Grundlage des vite+react+typescript Templates.

Dafür werden die minimalen Voraussetzungen für ein MapComponents-Projekt geschaffen.

```
→ test npm create mapcomponents-app@latest myApp
> npx
> "create-mapcomponents-app" myApp

Project cloned to myApp
→ test cd myApp
→ myApp npm i
npm warn deprecated lodash.isequal@4.5.0: This package is deprecated. Use require('node:util').isDeepStrictEqual instead.

added 751 packages, and audited 752 packages in 12s

→ myApp npm run dev
> app_title@0.0.0 dev
> vite

VITE v7.1.5 ready in 161 ms

→ Local: http://localhost:5173/
→ Network: use --host to expose
→ press h + enter to show help
```



Praktischer Teil



Ausliefern einer MapComponents Anwendung (mapcomponents vite Template)

- ▶ "yarn build" (oder "npm run build") erzeugt eine, für den Produktiveinsatz, optimierte Version der Anwendung
- ▶ "docker build" erzeugt ein Einsatzbereites Webserver Image



Wichtige Links

- ▶ **MapComponents Storybook** <https://mapcomponents.github.io/react-map-components-maplibre>
- ▶ **MapComponents Webseite** <https://mapcomponents.org>
- ▶ **MapComponents Organisation auf Github** <https://github.com/mapcomponents>
- ▶ **@mapcomponents/react-maplibre Repository** <https://github.com/mapcomponents/react-map-components-maplibre>