Mapbender Dokumentation

Release 4.0.0

Das Mapbender Team

Jun 30, 2025

CONTENTS

1 Installation 1

2 Mapbender Schnellstart: Erste Schritte mit Mapbender 19

3 Elemente 37

4 Backend 247

5 Anpassungen / Tipps & Tricks 275

6 FAQ - Häufig gestellte Fragen 305

7 Architektur 313

8 Entwicklung 319

9 Versionshistorie 327

10 QGIS2Mapbender Plugin 351

CHAPTER

ONE

INSTALLATION

Auf dieser Seite findet sich eine Übersicht über die unterschiedlichen Installationsanleitungen von Mapbender.

Allgemeine Installationsanleitung und Update

1.1 Installation auf Ubuntu/Debian

Die mitgelieferte SQLite-Datenbank ist für Testinstallationen geeignet. In dieser Datenbank befinden sich bereits vorkonfigurierte Demoanwendungen (sie liegt unter *<mapbender>/var/db/demo.sqlite*). Eine Anleitung für eine Testinstallation auf Basis des Symfony Webservers finden Sie unter *Aufruf von Mapbender unter Verwendung des lokalen Symfony Webservers*.

Hint: Für den Produktiveinsatz wird PostgreSQL dringend empfohlen. Wichtige Konfigurationshinweise dazu finden Sie im Kapitel *Optional > Mapbender Einrichtung auf PostgreSQL*.

1.1.1 Voraussetzungen

- PHP >= 8.1
- **Apache Installation mit folgenden aktivierten Modulen:**
  - mod_rewrite
  - libapache2-mod-php
- **alternativ: nginx-Installation mit folgenden aktivierten Modulen:**
  - php-fpm
- **PostgreSQL Installation**
  - Es wird empfohlen, eine PostgreSQL Datenbank für Mapbender zu verwenden.
  - Es wird empfohlen, einen eigenen Datenbankbenutzer für den Zugriff auf die Mapbender Datenbank anzulegen.

1.1.2 Vorbereitung

Installation der benötigten PHP-Extensions:

```
sudo apt install php-gd php-curl php-cli php-xml php-sqlite3 sqlite3 php-apcu php-intl openssl php-zip
↪php-mbstring php-bz2
```

- Bitte prüfen Sie die *FAQ - Häufig gestellte Fragen* für weitere PHP-Einstellungen.
- Bitte prüfen Sie die *API* zur Verwendung der Mapbender API.

1.1.3 Entpacken und im Webserver registrieren

Download der aktuellen Mapbender Version und entpacken nach */var/www/mapbender* oder ein anderes Verzeichnis:

```
wget https://mapbender.org/builds/mapbender-starter-current.tar.gz -O /var/www/mapbender-starter-
↪current.tar.gz
tar -zxf /var/www/mapbender-starter-current.tar.gz -C /var/www
mv $(ls -d /var/www/*/ | grep mapbender) /var/www/mapbender/
```

1.1.4 Konfiguration Apache 2.4

Datei */etc/apache2/sites-available/mapbender.conf* mit dem folgenden Inhalt anlegen:

```
Alias /mapbender /var/www/mapbender/public/
<Directory /var/www/mapbender/public/>
 Options MultiViews FollowSymLinks
 Require all granted

 # SetEnvIf aktiviveren, wenn die Mapbender API verwendet werden soll
 # SetEnvIf Authorization "(.*)" HTTP_AUTHORIZATION=$1

 RewriteEngine On
 RewriteBase /mapbender/
 RewriteCond %{REQUEST_FILENAME} !-f
 RewriteRule ^(.*)$ index.php [QSA,L]
</Directory>
```

Aktivieren der Seite und Apache neu starten:

```
a2ensite mapbender.conf
service apache2 reload
```

1.1.5 Konfiguration nginx

Alternativ zu Apache kann auch nginx verwendet werden. Dafür muss eine eigene Konfiguration in `/etc/nginx/sites-available` angelegt werden:

```
server {
    listen 80;
    listen [::]:80;
    server_name mapbender.localhost # bitte anpassen

    # SSL-Konfiguration wird hier empfohlen sofern der Mapbender nicht lokal ausgeführt wird

    root /var/www/mapbender/application/public;

    index index.php;

    location / {
        # Versuche zunächst als Datei auszuliefern, dann als Verzeichnis,
        # alles andere wird an die index.php weitergeleitet
        try_files $uri $uri/ /index.php$is_args$args;
    }

    # PHP-Skripte an den FastCGI server weitergeben
    location ~ \.php$ {
        include snippets/fastcgi-php.conf;
        fastcgi_pass unix:/run/php/php8.3-fpm.sock; # bitte anpassen, wenn eine andere PHP-Version
↪verwendet wird
    }

    # Zugriff auf Apache-Konfiugurationsdateien verbieten
    location ~ /\.ht {
        deny all;
    }
}
```

Aktivieren der Seite und nginx neu starten:

```
ln -s /etc/nginx/sites-available/mapbender /etc/nginx/sites-enabled/
systemctl restart nginx
```

1.1.6 Verzeichnisrechte

```
sudo chown -R :www-data /var/www/mapbender

sudo chmod -R ug+w /var/www/mapbender/var/log
sudo chmod -R ug+w /var/www/mapbender/var/cache
sudo chmod -R ug+w /var/www/mapbender/public/uploads

sudo chmod -R ug+w /var/www/mapbender/var/db/demo.sqlite
```

1.1.7 Nächste Schritte

Es kann nun auf die Mapbender Installation unter `http://[hostname]/mapbender/` zugegriffen werden.

Per Voreinstellung lauten die Anmeldedaten

- Benutzername: root
- Passwort: root

Zur Überprüfung der Konfiguration dient der folgende Befehl:

```
bin/console mapbender:config:check
```

Hint: Bitte beachten Sie, dass der Befehl `mapbender:config:check` die PHP-CLI Version nutzt. Die Einstellungen der CLI-Version können sich von denen der Webserver PHP-Version unterscheiden. Nutzen Sie beispielsweise `php -r 'phpinfo();'` zur Ausgabe der PHP-Webserver Einstellungen.

Glückwunsch! Mapbender wurde erfolgreich installiert. Informationen zu den ersten Schritten mit Mapbender finden sich im *Mapbender Schnellstart*.

1.1.8 Optional

LDAP

Zur Nutzung der optionalen LDAP-Anbindung wird zunächst die PHP-LDAP-Extension benötigt:

```
sudo apt install php-ldap
```

Note: Zur Verwendung von LDAP muss im Anschluss außerdem das LDAP-Bundle in Mapbender integriert werden. Weitere Anweisungen zur Einrichtung finden Sie in der README.md.

Mapbender Einrichtung auf PostgreSQL

Für den Einsatz in einer Produktivumgebung wird nachfolgend die Konfiguration einer PostgreSQL Datenbank beschrieben.

Voraussetzungen:

- Installation von PostgreSQL
- vorhandene Datenbank zur Mapbender-Konfiguration
- ggf. eigenen Benutzer für den Zugriff

Installation PHP-PostgreSQL Treiber:

```
sudo apt install php-pgsql php_pdo_pgsql
```

Die Konfiguration der Datenbankverbindung erfolgt über eine Variable, die den gesamten Verbindungsstring enthält. Konfigurieren Sie sie, indem Sie sie in Ihrer *.env.local*-Datei hinzufügen.

```
MAPBENDER_DATABASE_URL="postgresql://dbuser:dbpassword@localhost:5432/dbname?serverVersion=17&
↪charset=utf8"
```

Anlegen der Mapbender Datenbank, sofern diese noch nicht vorliegt:

```
cd /var/www/mapbender
bin/console doctrine:database:create
```

Einrichtung der Mapbender Tabellenstruktur und laden der Demo-Anwendungen:

```
cd /var/www/mapbender
bin/console doctrine:schema:create
bin/console mapbender:database:init -v
bin/composer run reimport-example-apps
```

Root-Benutzer für Zugriff anlegen:

```
bin/console fom:user:resetroot
```

Weitere Informationen zur Konfiguration im Kapitel *Details zur Konfiguration von Mapbender*.

1.2 Installation unter Windows

Mapbender benötigt eine Datenbank zur Speicherung der Administrationsinformation. Das Installationspaket enthält bereits eine SQLite-Datenbank, die direkt verwendet werden kann. Für den produktiven Einsatz wird allerdings die Nutzung einer PostgreSQL Datenbank empfohlen.

1.2.1 Voraussetzungen

- PHP NTS >= 8.1 (https://windows.php.net/download/)
- **Apache Installation, als Dienst eingerichtet (https://www.apachelounge.com/download/) mit folgenden aktiv**
  - mod_rewrite
  - mod_fcgid
- **PostgreSQL Installation (https://www.enterprisedb.com/downloads/postgres-postgresql-downloads)**
  - Es wird empfohlen, eine PostgreSQL Datenbank für Mapbender zu verwenden.
  - Es wird empfohlen, einen eigenen Datenbankbenutzer für den Zugriff auf die Mapbender Datenbank anzulegen.

Als Webserver kann auch Nginx verwendet werden, der für Windows aber als Beta-Version betrachtet wird. Sollte gewünscht sein, nginx auf Windows zu verwenden, kann die gleiche Konfiguration wie für die Linux-Installation verwendet werden.

1.2.2 Konfiguration PHP

- Entpacken des Zip-Archives, z.B. nach *c:\php*.
- Abhängig von der PHP-Version werden unter Windows PHP-Variablen für ein Temp-Verzeichnis nicht richtig gesetzt.
- Es muss deshalb geprüft werden, ob die folgenden Variablen (*php.ini*) gesetzt sind:

```
sys_temp_dir
upload_tmp_dir
date.timezone
```

- Der Pfad vom *PHP-bin*-Verzeichnis zur PATH-Variable (Windows-Umgebungsvariable) muss hinzugefügt werden.
- Aktivieren Sie die benötigten PHP-Erweiterungen in der *php.ini* Konfigurationsdatei:

```
# php.ini
extension=php_curl
extension=php_fileinfo
extension=php_gd
extension=php_intl
extension=php_pdo_pgsql
extension=php_pdo_sqlite
extension=php_pgsql
extension=php_openssl
extension=php_mbstring
extension=php_zip
extension=php_bz2
```

- Bitte prüfen Sie die *FAQ - Häufig gestellte Fragen* für weitere PHP-Einstellungen
- Bitte prüfen Sie die *API* zur Verwendung der Mapbender API

1.2.3 Mapbender entpacken und im Webserver registrieren

Download der aktuellen Mapbender Version und entpacken nach *c:\mapbender*

1.2.4 Konfiguration Apache

- Ein Unterordner *conf.d* muss im Verzeichnis *<apache>/conf* erstellt werden

In der *httpd.conf* am Ende einfügen:

```
# Include directory conf.d
Include "conf/conf.d/*.conf"
```

Datei *<apache>\conf\conf.d\mapbender.conf* mit dem folgenden Inhalt anlegen:

```
Alias /mapbender c:/mapbender/public/
<Directory c:/mapbender/public/>
 Options MultiViews FollowSymLinks
 Require all granted

 # SetEnvIf aktiviveren, wenn die Mapbender API verwendet werden soll
```

(continues on next page)

(continued from previous page)

```
 # SetEnvIf Authorization "(.*)" HTTP_AUTHORIZATION=$1

 RewriteEngine On
 RewriteBase /mapbender/
 RewriteCond %{REQUEST_FILENAME} !-f
 RewriteRule ^(.*)$ index.php [QSA,L]
</Directory>
```

Der Apache Webserverdienst muss im Anschluss neu gestartet werden.

1.2.5 mod_fcgid

Datei *<apache>\conf\conf.d\fcgi.conf* mit dem folgenden Inhalt anlegen:

```
LoadModule fcgid_module modules/mod_fcgid.so

FcgidInitialEnv PHPRC "c:/php/"
FcgidInitialEnv PATH "c:/php;C:/WINDOWS/system32;C:/WINDOWS;C:/WINDOWS/System32/Wbem"
FcgidInitialEnv SystemRoot "C:/Windows"
FcgidInitialEnv TEMP "C:/WINDOWS/TEMP"
FcgidInitialEnv TMP "C:/WINDOWS/TEMP"
FcgidInitialEnv windir "C:/WINDOWS"

FcgidPassHeader Authorization
FcgidIOTimeout 1200
FcgidConnectTimeout 1200
FcgidBusyScanInterval 1200
FcgidBusyTimeout 1200
FcgidErrorScanInterval 1200
FcgidIdleScanInterval 1200
FcgidIdleTimeout 1200
FcgidZombieScanInterval 1200
FcgidMaxProcesses 1000
FcgidOutputBufferSize 64
FcgidProcessLifeTime 3600
FcgidMaxRequestsPerProcess 10000
FcgidMinProcessesPerClass 0
FcgidFixPathinfo 0
MaxRequestLen 200000

<Files ~ "\.php$">
    Options Indexes FollowSymLinks ExecCGI
    AddHandler fcgid-script .php
    FcgidWrapper "c:/php/php-cgi.exe" .php
</Files>
```

1.2.6 Konfiguration PostgreSQL

Für den Einsatz in einer Produktivumgebung wird nachfolgend die Konfiguration einer PostgreSQL Datenbank beschrieben.

Voraussetzungen:

- Installation von PostgreSQL
- vorhandene Datenbank zur Mapbender-Konfiguration
- ggf. eigenen Benutzer für den Zugriff

Installation PHP-PostgreSQL Treiber:

```
# php.ini
extension=php_pgsql
extension=php_pdo_pgsql
```

Die Konfiguration der Datenbankverbindung erfolgt über eine Variable, die den gesamten Verbindungsstring enthält. Konfigurieren Sie sie, indem Sie sie in Ihrer *.env.local*-Datei hinzufügen.

```
MAPBENDER_DATABASE_URL="postgresql://dbuser:dbpassword@localhost:5432/dbname?serverVersion=17&
↪charset=utf8"
```

Weitere Informationen zur Einrichtung von Datenbankverbindungen finden sich im Kapitel *YAML Konfiguration (Konfigurations- und Anwendungsdateien)*.

Öffnen Sie nun die Eingabeaufforderung.

Zur Erzeugung der Datenbank geben Sie bitte folgenden Befehl ein (nur wenn die Datenbank nicht bereits vorliegt):

```
cd c:\mapbender
php.exe bin/console doctrine:database:create
```

Einrichtung der Mapbender Tabellenstruktur und laden der Demo-Anwendungen:

```
cd c:\mapbender
php.exe bin/console doctrine:schema:create
php.exe bin/console mapbender:database:init -v
php.exe bin/composer run reimport-example-apps
```

Für die Administration von Mapbender wird ein Root-Benutzer benötigt. Dieser Benutzer wird über den folgende Befehl angelegt:

```
php.exe bin/console fom:user:resetroot
```

Weitere Informationen zur Konfiguration im Kapitel *Details zur Konfiguration von Mapbender*

1.2.7 Der erste Start

Die Mapbender Installation kann unter **http://[hostname]/mapbender/** aufgerufen werden.

Überprüfung

Überprüfen Sie, ob der Alias erreichbar ist:

- `http://localhost/mapbender/`

Per Voreinstellung lauten die Anmeldedaten (wenn die SQLite-Datenbank verwendet wurde)

- Benutzername: root
- Passwort: root

Zur Überprüfung der Konfiguration dient der folgende Befehl:

```
php.exe bin/console mapbender:config:check
```

Hint: Bitte beachten Sie, dass der Befehl `mapbender:config:check` die PHP-CLI Version nutzt. Die Einstellungen der CLI-Version können sich von denen der Webserver PHP-Version unterscheiden. Nutzen Sie beispielsweise `php -r 'phpinfo();'` zur Ausgabe der PHP-Webserver Einstellungen.

Weitere Informationen dazu finden Sie unter *bin/console mapbender:config:check*.

Glückwunsch! Mapbender wurde erfolgreich installiert. Informationen zu den ersten Schritten mit Mapbender finden sich im *Mapbender Schnellstart*.

1.3 Aktualisierung von Mapbender auf eine neuere Version

Um Mapbender zu aktualisieren, müssen Sie die folgenden Schritte durchführen:

- Laden Sie die neueste Version herunter.
- Sichern Sie Ihre **Konfigurationsdateien** (*parameters.yml* für Versionen < 4.0.0 bzw. *.env.local* für Versionen >= 4.0.0, und die Dateien im Verzeichnis *config/packages*) und Ihre alte Mapbender Version (Dateien und Datenbank).
- Ersetzen Sie die Dateien durch die neuen Mapbender-Dateien.
- Vergleichen Sie die Konfigurationsdateien und prüfen diese auf neue Parameter und Änderungen.

Warning: Wenn Sie auf die Mapbender-Version 4 umsteigen, beachten Sie vor dem Upgrade der Datenbank unbedingt den *Hinweis im Migrationsleitfaden*!

- Aktualisieren Sie Ihre *Mapbender-Datenbank*.
- Übernahme Ihrer **Screenshots**: Kopieren Sie die Dateien Ihrer alten Mapbender Version von */public/uploads/* (*web/uploads* für Versionen < 4.0.0) in das */public/uploads*-Verzeichnis Ihrer neuen Mapbender Version.
- Wenn Sie Ihre eigenen **Templates** verwenden sollten, müssen Sie diese mit denen der neuen Version vergleichen (kam es zu Änderungen?).
- **Druck Templates**: Kopieren Sie eventuell selbst erstellte Drucktemplates zurück unter *config/MapbenderPrintBundle/templates/*.

- Importieren Sie die **Demo-Anwendungen**: über den Befehl `bin/composer run reimport-example-apps` oder über die Web-Administration, um sich den neusten Stand der Entwicklungen anzuschauen.
- Unter *Entpacken und im Webserver registrieren* ist beschrieben, wie die Konfigurationsdatei für den Apache Alias aussehen sollte.

Das war's auch schon! Schauen Sie sich Ihre neue Mapbender Version an.

Hint: Folgen Sie bitte zusätzlich den *Migrationshinweisen für bestimmte Versionen*.

1.3.1 Aktualisierungsbeispiel für Linux

Im Folgenden sind die einzelnen Schritte als Befehle aufgeführt.

```
# Laden Sie die neue Version herunter
wget https://mapbender.org/builds/mapbender-starter-current.tar.gz /tmp/build_mapbender/

tar xfz /tmp/build_mapbender/mapbender-starter-current.tar.gz

# Sichern Sie die alte Version
mv /var/www/mapbender /var/www/mapbender_save

# Aktivieren Sie den Code der neuen Version
cp -R /tmp/build_mapbender/mapbender-starter-v4.0.0 /var/www/
mv /var/www/mapbender-starter-v4.0.0 /var/www/mapbender

# Übernehmen Sie die Konfigurationsdateien in die neue Version von Mapbender
# Übernehmen Sie die Informationen von der config/parameters.yml nach .env.local bzw. /config/
↪parameters.yaml
# Übernehmen Sie die Informationen die Information der alten config.yml nach /var/www/mapbender/config/
↪packages/doctrine.yaml

# Händisch müssen Sie nun die Konfigurationsdateien auf neue Parameter überprüfen.
# Vergleichen Sie die Dateien parameters.yaml,packages/doctrine.yaml und ggf. andere Dateien aus dem
↪packages-Verzeichnis
# Sofern Sie eigene Templates angelegt haben, vergleichen Sie diese mit der neuen Mapbender Version.
# Sofern Sie Vorschaubilder hochgeladen haben: Kopieren Sie diese von der alten Version wieder nach
↪mapbender/public/uploads.
# Sofern Sie eigene Druckvorlagen verwenden: Kopieren Sie diese wieder nach config/MapbenderPrintBundle/
↪templates/.

# Setzen Sie die Schreibrechte für Besitzer (u), Gruppe (g) und Andere (a). Weisen Sie die Skripte der
↪Apache Gruppe (www-data) zu.
sudo chmod -R ugo+r /var/www/mapbender
sudo chown -R :www-data /var/www/mapbender

# Aktualisieren Sie Ihre Mapbender Datenbank
cd /var/www/mapbender/
bin/console doctrine:schema:update --complete --dump-sql
bin/console doctrine:schema:update --complete --force

# Importieren Sie die Demo-Anwendungen, um sich den neusten Stand der Entwicklungen anzuschauen
bin/composer run reimport-example-apps

# Aufbau der symbolischen Links
```

(continues on next page)

(continued from previous page)

```
bin/console assets:install public --symlink --relative

# Setzen Sie die Schreibrechte für Besitzer (u), Gruppe (g) und Andere (a). Weisen Sie die Skripte der
↪Apache Gruppe (www-data) zu.
sudo chmod -R ug+r /var/www/mapbender
sudo chown -R :www-data /var/www/mapbender

# Sie benötigen Schreibrechte für die Verzeichnisse var/cache und var/log.
sudo chmod -R ug+w /var/www/mapbender/var/cache
sudo chmod -R ug+w /var/www/mapbender/var/log
sudo chmod -R ug+w /var/www/mapbender/public/uploads
```

1.3.2 Aktualisierungsbeispiel für Windows

```
# Laden Sie die neue Version herunter https://mapbender.org/builds/

# Sichern Sie die alte Version (Dateien und Datenbank)

# Übernehmen Sie die Konfigurationsdateien in die neue Version von Mapbender.
# Vorher müssen Sie diese händisch auf neue Parameter und Änderungen überprüfen.

# Rufen Sie die bin/console Befehle über die php.exe auf.
# Hierzu müssen Sie ein Standardeingabefenster öffnen.
c:
cd mapbender

# Aktualisieren Sie Ihre Mapbender Datenbank
php.exe bin/console doctrine:schema:update --complete --dump-sql
php.exe bin/console doctrine:schema:update --complete --force

# Importieren Sie die Anwendungen aus der mapbender.yaml Datei, um sich den neusten Stand der
↪Entwicklungen anzuschauen
php.exe bin/composer run reimport-example-apps

# Ausspielen in den web-Bereich
php.exe bin/console assets:install public

# Löschen Sie den Cache und die Logdateien unter mapbender/var/cache und mapbender/var/log

# Sofern Sie eigene Templates angelegt haben, vergleichen Sie diese mit der neuen Mapbender Version.
# Sofern Sie Vorschaubilder hochgeladen haben: Kopieren Sie diese von der alten Version wieder nach
↪mapbender/public/uploads.
# Sofern Sie eigene Druckvorlagen verwenden: Kopieren Sie diese wieder nach config/MapbenderPrintBundle/
↪templates/
```

Installationsanleitung und Details für die Mapbender-Entwicklung

1.4 Aufruf von Mapbender unter Verwendung des lokalen Symfony Webservers

Der Aufruf von Mapbender im lokalen Symfony Webserver ermöglicht eine schnelle Konfiguration mithilfe von Symfony-eigenen Mechanismen. Das ermöglicht Ihnen einen schnellen Test, ohne eine Integration in einen eigens aufgesetzen Webserver vorzunehmen.

Note: Das lokale Webserver-Bundle wurde entfernt und durch den lokalen Symfony Web Server ersetzt. Installiere das Symfony CLI Entwicklungswerkzeug, um diesen zu nutzen.

Hint: Der Symfony-eigene Webserver eignet sich nicht für die Produktivumgebungen.

In dieser Anleitung wird die im Installationspaket mitgelieferte SQLite-Datenbank verwendet.

- Bitte prüfen Sie die Systemvoraussetzungen unter *Linux* bzw. *Windows*.
- Laden Sie den aktuellen *Mapbender-Build <https://mapbender.org/builds/>* herunter.
- Entpacken in die Dateien ein beliebiges Verzeichnis.
- Starten Sie den installierten lokalen Symfony Webserver.

```
symfony server:start --no-tls
```

Der Befehl führt einen lokalen Webserver aus, sodass Mapbender nun auf dem lokalen Rechner erreichbar ist und die Konsole den Status des Servers mitprotokolliert.

Note: Beachten Sie, dass über diesen Aufruf die Anwendungsumgebung (`prod` oder `dev`) definiert werden kann:

```
APP_ENV=prod symfony server:start --no-tls
```

Der gewünschte Server-Port kann auch als Argument mitgegeben werden (siehe unten). Weitere Optionen werden ausgegeben, wenn der `--help` Parameter mitgegeben wurde.

```
symfony server:start --no-tls -port=8002
```

```
[OK] Web server listening
     The Web server is using PHP CLI 8.2.10
     http://127.0.0.1:8002
```

1.5 Git-basierte Installation

Note: Die Git-basierte Installation ist für die Entwicklung von Mapbender gedacht oder für projektspezifische Anpassungen, die in einem geklonten Mapbender-Starter abgelegt werden können. Dort können Sie ihre eigenen Templates, Ihre eigenen Anwendungen oder eigene Bundles hinterlegen. Über den Composer laden Sie dann Mapbender und alle weiteren genutzten Module (z.B. Digitizer) in der Version ein, die sie benötigen. Bei einem Mapbender-Update vergleichen Sie den Stand der Mapbender-Starter-Repositories. Die grundsätzlichen Anpassungen sind die Versionen in der *composer.json*.

Alle wesentlichen Informationen werden vom Mapbender-Team auf GitHub dokumentiert. Die zentralen Dokumentationsdateien und ihre Inhalte lauten:

- README.md beschreibt die Installation aus den Git-Quellen des Mapbender-Starter Git-Repository.
- CHANGELOG.md listet Versionsänderungen der Mapbender-Entwicklungsstände auf.
- UPGRADING.MD gibt wichtige Hinweise zu Mapbender-Installationsupgrades auf bestimmte Versionen.
- Der Contributing Guide stellt die Software-Architektur und die Schritte zur Entwicklung von Mapbender vor.
- Die /docs/-README.md dient als Einstiegsseite für die technische Dokumentation von Mapbender.

1.6 Details zur Konfiguration von Mapbender

1.6.1 Konfigurationsschritte

Im Folgenden werden die für die Mapbender-Installation aufgeführten Konfigurationsschritte von Mapbender näher erläutert. Es sind sechs Schritte notwendig:

- Erzeugen der Datenbank
- Erzeugen der Datenbankschemas
- Kopieren des bundle Assets in das öffentlich zugängliche *public*-Verzeichnis
- Erzeugen des “root” Benutzers
- Initialisieren der Datenbank
- Laden der Demo-Anwendungen in die Datenbank

Diese Schritte werden mit dem console-Hilfsprogramm des Symfony Frameworks durchgeführt, auf dem Mapbender aufbaut. Hier noch einige wichtige Hinweise, bevor Sie fortfahren:

Note: **Hinweis:** Das console-Hilfsprogramm wird Dateien in die Verzeichnisse var/cache und var/log schreiben. Für diese Operationen werden die Benutzerrechte des Benutzers benötigt, mit dem Sie angemeldet sind. Sie benötigen ebenfalls Benutzerrechte für das Verzeichnis var/db und die SQLite Datenbank. Wenn Sie die Applikation in Ihrem Browser öffnen, wird die SQLite-Datenbank mit Server-PHP-Prozess versuchen, auf diese Dateien zuzugreifen oder in die Verzeichnisse mit anderen Benutzerrechten zu schreiben. Stellen Sie sicher, dass Sie den Verzeichnissen und Dateien Schreib- und Leserechte zugewiesen haben.

Note: **Wichtiger Hinweis:** Die folgenden `bin/console` Schritte gehen davon aus, dass Sie sich oberhalb des *bin*-Verzeichnisses befinden (für die git-Installation bedeutet das *mapbender/application/* , andernfalls *mapbender/*).

```
cd mapbender/
oder für die git-basierte Installation
cd mapbender/application
```

Anpassen der Konfigurationsdatei

Die Parameter der Datenbankverbindung sind in der Datei *.env.local* gespeichert. Weitere Konfigurationsparameter finden sich in der Datei *parameters.yaml* und anderen Dateien im *config*-Verzeichnis.

Mehr Informationen dazu finden Sie im Kapitel : *YAML Konfiguration (Konfigurations- und Anwendungsdateien)*.

Erzeugen der Datenbank

Hint: Generell wird empfohlen, das Erzeugen der Datenbanken extern, etwa über ein graphisches Datenbank-Werkzeug wie bspw. pgAdmin vorzunehmen.

Alternativ kann die Datenbank mit Symfony erzeugt werden. Beachten Sie, dass dazu die benötigten Datenbank-Benutzerrechte vorliegen müssen.

Rufen Sie dazu folgenden Befehl mit dem console-Hilfsprogramm auf:

```
bin/console doctrine:database:create
```

Erzeugen des Datenbankschemas

Erzeugen des Datenbankschemas über Symfony:

```
bin/console doctrine:schema:create
```

Kopieren des Asset Bundles

Jedes Bundle hat seine eigenen Abhängigkeiten - CSS-Dateien, JavaScript-Dateien, Bilder und mehr – diese müssen in das öffentliche *public*-Verzeichnis kopiert werden:

```
bin/console assets:install public
```

Sie können auch einen symbolischen Link verwenden, statt die Dateien zu kopieren. Dies erleichtert die Bearbeitung der abhängigen Dateien in den *bundle*-Verzeichnissen.

```
bin/console assets:install public --symlink --relative
```

Erzeugen des administrativen Benutzers

Der erste Benutzer, der alle Privilegien hat, wird mit folgendem Kommando erzeugt:

```
bin/console fom:user:resetroot
```

Dieses Kommando wird interaktiv alle notwendigen Informationen abfragen und den Benutzer in der Datenbank erzeugen.

Sie können auch den Modus `silent` verwenden, wenn Sie ein Skript nutzen möchten, um Mapbender zu installieren und dabei nicht nach Parametern gefragt werden wollen.

```
bin/console fom:user:resetroot --username="root" --password="root" --email="root@example.com" --silent
```

Initialisieren der Datenbank

Führen Sie das nachfolgende Kommando aus, um die Datenbank zu initialisieren und startbereit zu machen:

```
bin/console mapbender:database:init
```

Importieren von Anwendungen

Sie können die Anwendungen, die in dem Ordner *applications* (innerhalb von *config/*) definiert sind, in die Datenbank importieren:

```
bin/composer run reimport-example-apps
```

1.6.2 Konfigurationsdateien

Die Konfigurationsdateien liegen im Verzeichnis *config/*.

Mehr Informationen dazu finden Sie im Kapitel: *YAML Konfiguration (Konfigurations- und Anwendungsdateien)*.

Außerdem die Datei *env.local* und *.env* im Verzeichnis *application*.

1.6.3 Produktions- und Entwicklungsumgebung und Caches

Mapbender bietet zwei Umgebungen an: eine Produktionsumgebung für den normalen Betrieb und eine Entwicklungsumgebung, in dem die Anwendungen getestet werden können. Dieses Konzept orientiert sich an den "Environments" im Symfony Framework.

Die Produktionsumgebung wird mit der URL http://localhost/index.php aufgerufen, die Entwicklungsumgebung mit der URL http://localhost/index_dev.php. Der Aufruf über `index_dev.php` kann und sollte nur vom localhost erfolgen. Sie können die Datei *index_dev.php* anpassen und IP-Adressen hinzufügen, so dass auch von diesen Adressen der Entwicklungsmodus erreichbar ist.

Es gibt Unterschiede im Verhalten von *index.php* und *index_dev.php*:

- Der Cache-Mechanismus verhält sich in der Entwicklungsumgebung anders: Es werden nicht alle Dateien gecacht, so dass vorgenommene Änderungen direkt sichtbar sind. Dadurch ist der Aufruf einer Anwendung über `index_dev.php` immer langsamer als im Produktivbetrieb.

Im Detail werden in der Entwicklungsumgebung von Mapbender u.a. die CSS, JavaScript und Übersetzungsdateien nicht gecacht.

In der Produktionsumgebung werden diese aber in *var/cache* abgelegt.

- In der Entwicklungsumgebung werden Fehlermeldungen und ihr Stacktrace direkt an der Oberfläche angezeigt. In der Produktionsumgebung werden die Fehlermeldungen in die Datei *var/log/prod.log* geschrieben.
- Die Entwicklungsumgebung zeigt den Symfony Profiler an. Dort werden Dinge protokolliert, die nur für die Entwickler, aber nicht für Außenstehende sichtbar sein sollten.

Das Verzeichnis *var/cache* enthält die einzelnen Cache-Dateien. Es werden Verzeichnisse für jede Umgebung (`prod` und `dev`) angelegt, das Verhalten des dev-Caches ist aber, wie angesprochen, anders.

Eine Umgebung kann über die Variable `APP_ENV` explizit festgelegt werden. Stellen Sie sicher, dass Sie dies auf `prod` ändern, wenn Sie Ihre Anwendung für die Öffentlichkeit bereitstellen. Der Wert kann auf verschiedene Arten geändert werden:

- durch Überschreiben des Werts in einer *.env.local*-Datei,
- durch Festlegen einer Umgebungsvariable in Ihrer Apache2-vHost-Konfiguration: `SetEnv APP_ENV prod`,
- durch explizites Festlegen beim Starten des lokalen Webservers:

```
APP_ENV=prod symfony server:start --no-tls
```

1.7 Migrations-Leitfaden

Hint: Diese Seite enthält Migrationsanweisungen für die Umstellung auf Mapbender 4. Migrationsanweisungen zu Versionen < 4.0.0 finden Sie in der englischen Version dieser Seite. Allgemeine Tipps zur Aktualisierung finden Sie stattdessen auf der Seite *Aktualisierung von Mapbender auf eine neuere Version*.

Tip: Für ausführliche Informationen des Mapbender-Entwicklungsteams siehe auch den Upgrading Guide auf GitHub.

1.7.1 Umstellung auf Mapbender 4.0.0

- Schauen Sie sich den Upgrading Guide auf GitHub sorgfältig an, bevor Sie das Update durchführen.
- Sie werden feststellen, dass sich die Symfony Verzeichnisstruktur stark verändert hat.
- Für Mapbender 4 wurden alle *.yml*-Dateierweiterungen in *.yaml* umgewandelt.
- Die *config.yml* ist entfallen.
- Öffentliche Dateien wurden von *web/* nach *public/* verschoben.
- Anstelle von mehreren Parametern in der Datei *parameters.yml* wird die Datenbankdefinition durch eine Umgebungsvariable `MAPBENDER_DATABASE_URL` ersetzt. Konfigurieren Sie sie, indem Sie diese in Ihrer *.env.local*-Datei hinzufügen. Wenn Sie mehrere Verbindungen haben, verwenden Sie eine Umgebungsvariable pro Verbindung und konfigurieren diese in der Datei *config/packages/doctrine.yaml*.

- Der Apache Virtual Host / ALIAS muss geändert werden. Verweisen Sie auf *public* (statt auf *web*). Rufen Sie `index.php` (anstelle von `app.php`) auf. Siehe *Installationsanleitung*.
- Die Umgebung kann nun über die Umgebungsvariable `APP_ENV` gesetzt werden (siehe *.env.local*). `index_dev.php` (vorher `app_dev.php`) ist weiterhin als Alternative für den Zugriff auf die Entwicklungsumgebung verfügbar.
- Datenbankberechtigungen können mit `bin/console mapbender:security:migrate-from-acl` migriert werden. Führen Sie dies aus, bevor Sie den Befehl `schema:update` ausführen, da sonst Ihre alten ACL-Tabellen nicht mehr vorhanden sind.

Upgrade der Datenbank

Warning: Führen Sie die folgenden Befehle exakt in der angegebenen Reihenfolge aus, um ein Upgrade durchzuführen (nachdem Sie die Verzeichnisstruktur von Symfony auf den neuesten Stand gebracht haben). Erstellen Sie zuerst ein Backup Ihrer Datenbank!

1. Dies ersetzt den aus doctrine entfernten `json_array`-Typ durch `json`. Wenn Sie ein anderes DBMS als SQlite, PostgreSQL und MySQL verwenden, müssen Sie dies manuell tun.

```
bin/console mapbender:database:upgrade
```

2. Migriert Sicherheitsdefinitionen aus dem ACL-System in das neue Rechtesystem:

```
bin/console mapbender:security:migrate-from-acl
```

3. Aktualisiert den Rest der Datenbank. Dies muss als letzter Schritt ausgeführt werden, da es die alten ACL-Tabellen löscht:

```
bin/console doctrine:schema:update --complete --force
```

CHAPTER

TWO

MAPBENDER SCHNELLSTART: ERSTE SCHRITTE MIT MAPBENDER

Mapbender ist ein webbasiertes Geoportal-Framework zum Veröffentlichen, Registrieren, Anzeigen, Navigieren und Überwachen von Diensten, mit der Möglichkeit, Dienste gesichert anzubieten. Administrationsoberflächen ermöglichen es Administratoren, Karten- und Datendienste zu verwalten und zu kategorisieren. Über die Administration können einzelnen Benutzern und Gruppen Zugriffe gewährt werden. Mapbender wird unter Verwendung moderner Webtechnologien entwickelt. Die Grundlage dazu bildet das Symfony-Framework. Auf der Clientseite wird OpenLayers genutzt.

Mit dieser Code-Grundlage wird die Idee eines Geoportal-Frameworks fortgesetzt. Zentrale Mapbender-Funktionen sind:

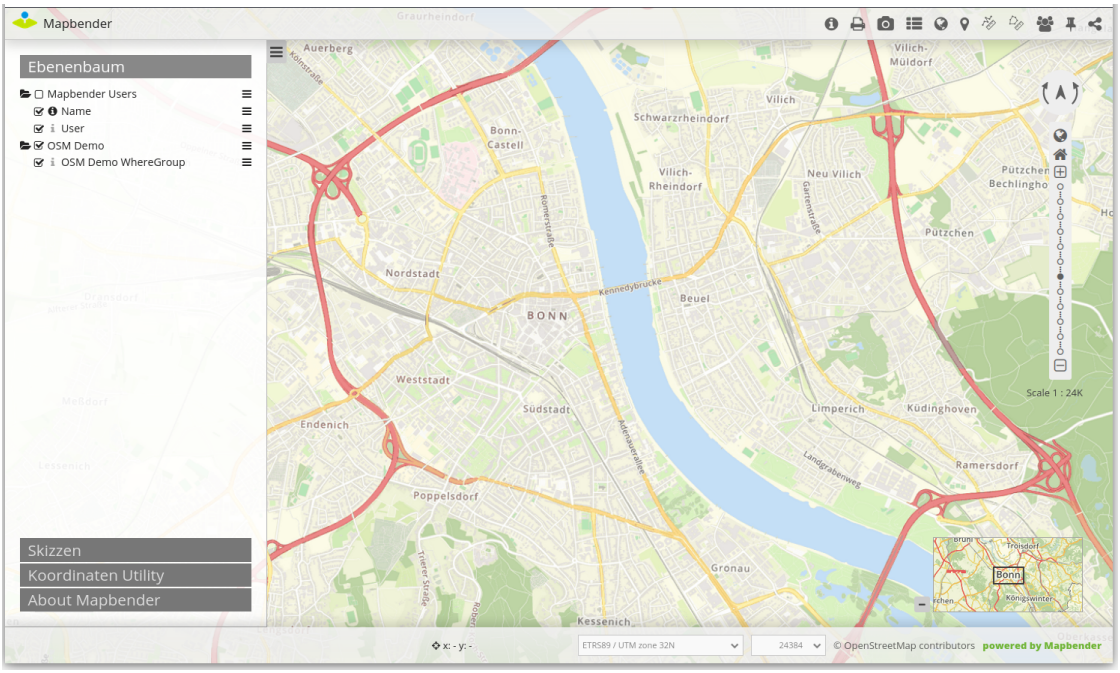
- Anwendungen können direkt im Browser erstellt, konfiguriert und gestylt werden.
- Dienste (wie WMS) können in einem Dienst-Repository verwaltet und mit Anwendungen verbunden werden.
- Das Rechtemanagement ist sowohl für einzelne Benutzer als auch Gruppen einfach zu verwalten, egal ob sie in einer Datenbank oder über LDAP gespeichert werden.
- Der Administrator braucht keine Zeile Code zu schreiben, da die Konfiguration über die webbasierte Administrationsoberfläche erfolgt.
- Suchen können konfiguriert werden.
- Anwendungen zur Digitalisierung können aufgebaut werden.
- Anwendungen können flexibel für Smartphones und Tablets angelegt werden.

Sie brauchen nichts weiter als einen Webbrowser für diesen Schnellstart.

Diese Schnellstart-Anleitung umfasst die folgenden Kapitel:

- *1. Mapbender starten*
- *2. Eine Anwendung erstellen*
- *3. Elemente in eine Anwendung einfügen*
- *4. Datenquellen (Sources) verwenden*
- *5. Benutzer- und Gruppenverwaltung*
- *6. Rechteverwaltung*
- *7. Anwendung beim Start positionieren*

Eine Mapbender Anwendung kann wie folgt aussehen:



2.1 Mapbender installieren

Dieser Schnellstart erklärt die Mapbender-Grundlagen nach erfolgter Installation und bietet einen schnellen Einstieg in die Mapbender-Oberfläche. Hinweise zur Installation von Mapbender finden Sie unter *Installation*.

2.2 1. Mapbender starten

1. Wählen Sie **Mapbender** aus dem Startmenü (sofern vorher eine solche browseröffnende Verknüpfung erstellt wurde) oder besuchen Sie http://localhost/mapbender (Adresse kann unter Umständen abweichen, je nachdem wie der Apache Alias in der Datei /etc/apache2/sites-available/mapbender.conf erstellt wurde, siehe auch *Installation*).
2. Das Mapbender-*Backend* sollte anschließend im Browserfenster erscheinen.

Falls Sie Schwierigkeiten haben Mapbender zu starten, überprüfen Sie, ob der Apache Web Server und die PostgreSQL-Datenbank korrekt funktionieren.

2.2.1 Starten von Mapbender als Produktivumgebung

Mapbender bietet zwei Umgebungen an: **dev** (Standard) und **prod**.

Die Entwicklungsumgebung zeigt vollständige Fehlermeldungen (einschließlich Stacktraces) im Browser und aktiviert die Symfony-Debug-Konsole und den Profiler. Außerdem wird das Caching deaktiviert. In der Produktionsumgebung wird das Caching aktiviert, zusätzlich werden nur allgemeine Fehlermeldungen angezeigt. Ausführlichere Meldungen werden hingegen in die Logdateien geschrieben.

Eine Umgebung kann über die Variable `APP_ENV` explizit festgelegt werden. Stellen Sie sicher, dass Sie dies auf `prod` ändern, wenn Sie Ihre Anwendung für die Öffentlichkeit bereitstellen. Der Wert kann auf verschiedene Arten geändert werden:

- durch Überschreiben des Werts in einer *.env.local*-Datei,
- durch Festlegen einer Umgebungsvariable in Ihrer Apache2-vHost-Konfiguration: `SetEnv APP_ENV prod`,
- durch explizites Festlegen beim Starten des lokalen Webservers:

```
APP_ENV=prod symfony server:start --no-tls
```

2.2.2 Das Backend

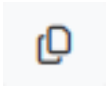
1. Nach dem erfolgreichen Start erscheint die Anwendungsübersicht des Mapbender-Backends. Das *Backend* listet nicht angemeldeten Besuchern die öffentlichen Anwendungen auf. Die Anwendungen werden mit Vorschaubild, Titel, URL-Titel und Beschreibung aufgelistet.
2. Eine Anwendung kann per Klick auf den Titel, das Vorschaubild oder über aufgerufen werden.
3. Um Mapbender verwalten zu können, muss zunächst eine Anmeldung in das *Backend* erfolgen. Klicken Sie auf den Link **Anmelden** oben rechts, um die Anmeldeseite aufzurufen. Sie können sich mit dem Benutzer anmelden, der während der Installation erzeugt wurde. Er kann **root** heißen mit dem Kennwort **root** - dies ist der Standardbenutzer/das Standardkennwort nach der Installation von Mapbender. Bitte ändern Sie das Kennwort, wenn Sie eine Produktivumgebung betreiben.

> **Warning:** Löschen Sie den Benutzer **root** nicht.

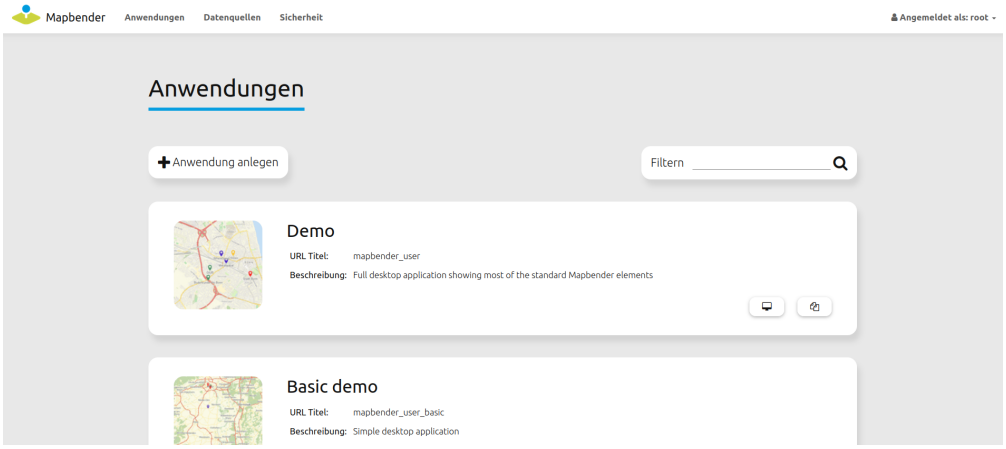
Nach erfolgreicher Anmeldung werden Sie zum Administrationsbackend weitergeleitet.

2.2.3 Anwendungsübersicht

In der Anwendungsübersicht finden Sie eine Liste mit allen verfügbaren Anwendungen. Für den Administrator stehen folgende Funktionen zur Verfügung:

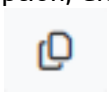
- Titel, URL-Titel und Beschreibung
- Vorschaubild der Anwendung
- Filtertextfeld zur Suche von Anwendungen
- Option zum Anlegen neuer Anwendungen
- Link zur Anwendung
-  Kopieren der Anwendung

- Exportieren der Anwendung
- Bearbeiten der Anwendung
- macht die Anwendung öffentlich zugänglich
- Löschen der Anwendung



2.3 2. Eine Anwendung erstellen

Es gibt drei verschiedene Möglichkeiten, durch die neue Anwendungen erstellt werden können:

Einerseits besteht die Option, diese aus bereits vorhandenen Anwendungen zu erstellen. Dies erfolgt über einen Klick auf den  Button in der Anwendungsübersicht. Die Applikation erhält dabei den gleichen Titel und URL-Titel zuzüglich dem Zusatz **_db** (bei yaml-Anwendungen) oder **_imp** (bei Anwendungen, die aus der Datenbank heraus kopiert werden). Alle zuvor definierten Elemente und Konfigurationen werden ebenfalls übernommen. Eine weitere Möglichkeit ist der Import einer Anwendung. Zusätzliche Informationen hierzu finden sich unter *YAML Konfiguration (Konfigurations- und Anwendungsdateien)*.

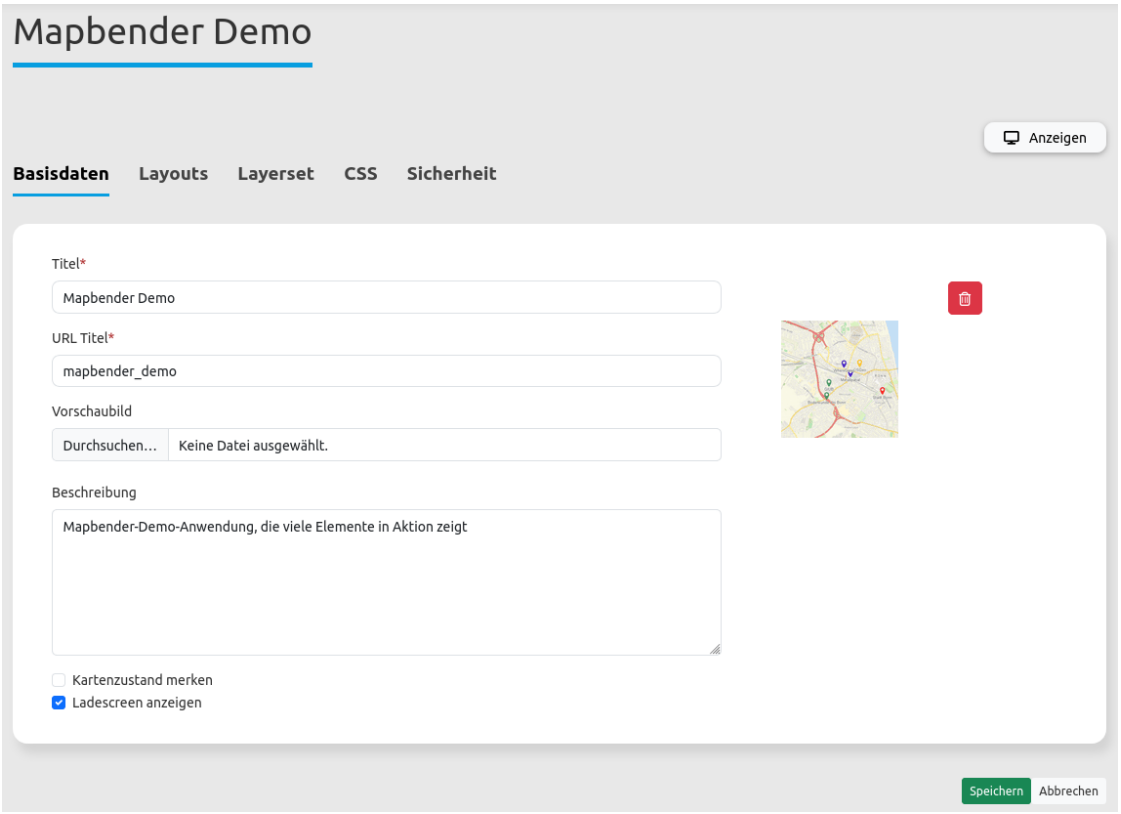
Es können außerdem komplett neue Anwendungen über das *Backend* definiert werden. Die einzelnen Arbeitsschritte hierfür werden im Folgenden näher erläutert:

1. Wählen Sie zunächst **Anwendung anlegen** auf der Anwendungsübersichtsseite.
2. Wählen Sie anschließend eine Vorlage für die Anwendung. Diese bestimmt den Aufbau der Anwendung. Zur Auswahl stehen: Fullscreen, Fullscreen alternative, Mapbender Mobile template. Es ist ebenfalls möglich, eigene Vorlagen anzulegen und neuen Anwendungen zuzuordnen.

Tip: Beachten Sie, dass Layout-, Icon- und Farbanpassungen online über den CSS-Editor und/oder lokal über Konfigurationsdateien erfolgen. Lesen Sie dazu die Dokumentation unter *Wie werden eigene Style-*

Vorlagen (templates) erzeugt?.

1. Geben Sie einen Titel, einen URL-Titel und ggf. eine Beschreibung für die Anwendung an. Titel und URL-Titel können identisch sein. Letzterer muss sich nach den Standards der festgelegten URL-Syntax richten: Üblicherweise sind Kleinbuchstaben, Zahlen sowie der Binde- und Unterstrich erlaubt.
2. Laden Sie ggf. eine Bilddatei hoch. Diese wird dann als Vorschaubild in der Anwendungsübersicht angezeigt. Klicken Sie dazu unter dem Abschnitt Vorschaubild auf den Button **Datei auswählen**.
3. Setzen Sie ein Häkchen bei **Kartenzustand merken**, um bestimmte Kartenparameter und -einstellungen persistent zu machen. Weitere Informationen finden Sie unter *Share-Elemente*.
4. Setzen Sie ein Häkchen bei **Ladescreen anzeigen**, um ein Bild bei Anwendungsstart anzuzeigen, das die Ladezeit graphisch überbrückt.
5. Klicken Sie **Speichern**, um die Anwendung zu erzeugen. Nach der Erstellung können Sie Elemente (z.B. Kartenelement, Navigation, Legende) und Dienste hinzufügen.



2.4 3. Elemente in eine Anwendung einfügen

Mapbender-Anwendungen bestehen aus den Bereichen Obere Werkzeugleiste, Sidepane, Kartenbereich und Fußzeile. In diese Bereiche kann eine Vielzahl von Elementen eingebunden werden.

1. Wählen Sie **Anwendungen** → **Layouts**.

2. Klicken Sie auf + neben einem Bereich, um einen Überblick über die dort verfügbaren Elemente zu erhalten.

Note: Beachten Sie dabei den jeweiligen Bereich Ihrer Anwendung. Es können nicht alle Elemente zu jedem Bereich hinzugefügt werden.

---

1. Wählen Sie ein Element aus der Liste aus. Beispielsweise ist die Übersicht (overview) nur in den Kartenbereich integrierbar.
2. Konfigurieren Sie das Element.

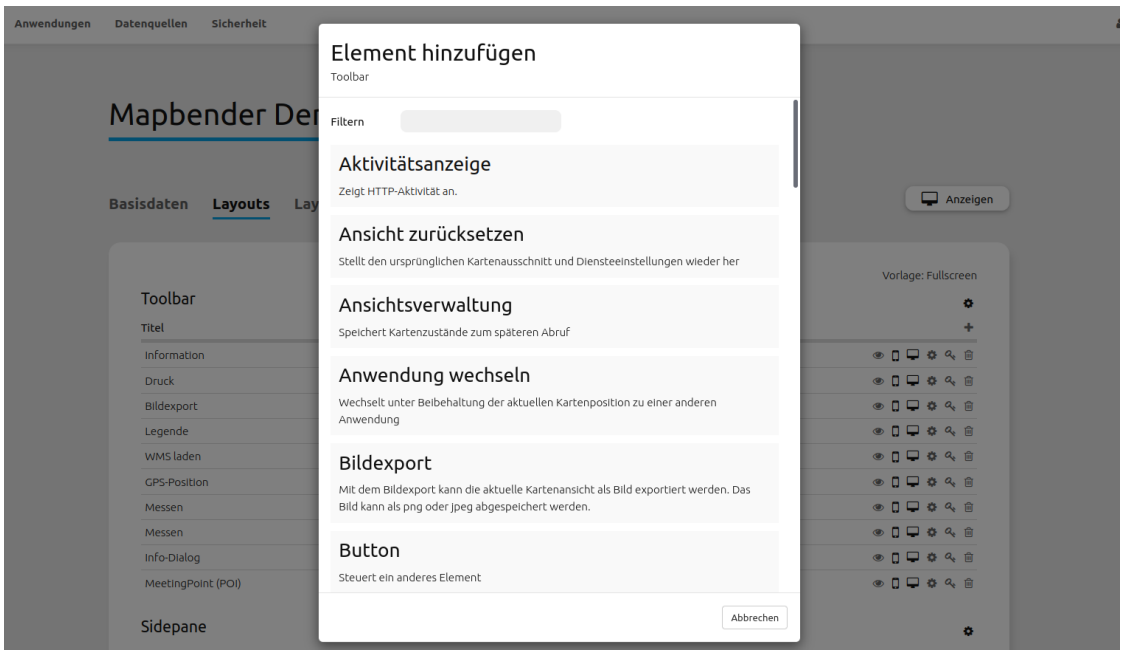
Note: Hinweis: Wenn Sie ein Element z.B. *Karte (Map)* auswählen, sehen Sie lediglich die Optionen für dieses Element und können es entsprechend konfigurieren.

---

1. Die Position von Elementen kann per Drag & drop innerhalb und zwischen Bereichen verändert werden.

2. Schauen Sie sich Ihre Anwendung an. Öffnen Sie Ihre Anwendung über den Button .

Jetzt sollten sie eine Idee davon haben, wie einfach es ist, eine Mapbender-Anwendung zu konfigurieren.



Im Folgenden finden Sie eine vollständige Liste aller Elemente inklusive ihrer Funktion. Detaillierte Informationen können Sie in den jeweiligen Kapiteln der *Mapbender Dokumentation* nachlesen.

- *Über-Mapbender-Dialog (About Dialog)*: Zeigt Informationen über Mapbender an.
- *Aktivitätsindikator (Activity Indicator)*: Zeigt die HTTP-Aktivität an.
- *Anwendung wechseln (Application Switcher)*: Wechselt unter Beibehaltung der aktuellen Kartenposition zu einer anderen Anwendung.
- *Hintergrund wechseln*: Ermöglicht den Wechsel zwischen selbst definierten Hintergrundkarten.
- *Button*: Bindet ein Element als Button ein.
- *Koordinaten Utility*: Transformiert Koordinaten und navigiert zu ihnen auf der Karte.
- *Koordinatenanzeige (Coordinates Display)*: Zeigt Mausposition in den Kartenkoordinaten an.
- *Copyright*: Zeigt Nutzungsbedingungen an.
- *Data Manager*: Erzeugt und speichert Sachinformationen in einer Datenbank.*
- *Digitizer*: Erzeugt und speichert Geometrieinformationen in einer Datenbank.*
- *Dimensions-Handler*: Bindet Dienste mit einer zeitlichen Dimension in die Anwendung ein.
- *Information (FeatureInfo)*: Gibt Informationen eines Dienstes aus.
- *GPS-Position*: Erzeugt einen Button zur Anzeige der eigenen GPS-Position.
- *HTML (Element)*: Bietet die freie Definition von HTML zur Einbindung von Bildern, Texten oder Links.
- *Bildexport (ImageExport)*: Exportiert einen Kartenausschnitt.
- *Ebenenbaum (Layertree)*: Auflistung aller eingebundenen Layersets und Layer samt Auswahl- und Sortiermöglichkeiten.
- *Legende (Legend)*: Zeigt die Legende von aktiven Diensten an.
- *Link*: Verlinkt zu einer externen URL.
- *Karte (Map)*: Erstellt ein zentrales Kartenelement, in welches die Layersets und Layer eingebunden sind.
- *Navigationswerkzeug (Navigation Toolbar)*: Ermöglicht die Navigation in der Karte über ein graphisches Steuerelement.
- *Übersichtskarte (Overview)*: Zeigt eine kleinere Übersichtskarte über der Hauptkarte an.
- *MeetingPoint (POI)*: Generiert einen Treffpunkt, welcher mit Hinweistexten und über eine URL verschickt wird.
- *Druck (PrintClient)*: Erzeugt einen Druckdialog, mit dem Karteninhalte als PDF exportiert und gedruckt werden können.
- *Linien-/Flächenmessung (Line/Area Ruler)*: Erlaubt das Messen von Linien und Flächen in der Karte.
- *Maßstabsanzeige (Scale display)*: Zeigt aktuellen Maßstab numerisch an.
- *Maßstabsauswahl (Scale selector)*: Dieses Element zeigt eine Auswahlbox mit Maßstäben an.
- *Maßstabsleiste (Scale bar)*: Zeigt den aktuellen Maßstab graphisch an.
- *Suchen (Search Router)*: Ermöglicht die Konfiguration von individuellen Suchen.
- *URL teilen (Share URL)*: Teilt die aktuelle Kartenansicht über eine URL.
- *Einfache Suche (SimpleSearch)*: Erstellt eine Einfeldsuche.

- *Skizzen (Sketch)*: Ermöglicht das Zeichnen verschiedener Formen in der Karte.
- *SRS Auswahl (SRS Selector)*: Generiert die Möglichkeit, eine Projektion (SRS) über eine Auswahlbox zu wechseln.
- *Ansichtsverwaltung (View Manager)*: Speichert Kartenzustände zum späteren Abruf.
- *WMS laden (WMS Loader)*: Lädt einen WMS per getCapabilities-Request.

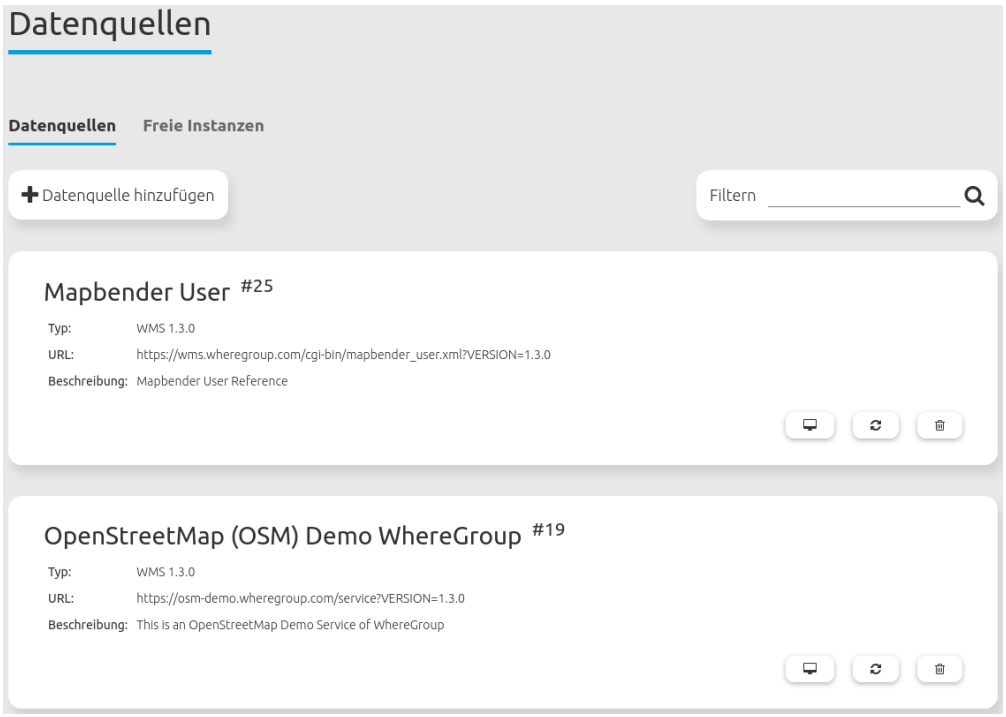
Hint: Mit einem * markierte Elemente benötigen zusätzliche Dependencies.

2.5 4. Datenquellen (Sources) verwenden

In Mapbender können Dienste vom Typ OGC WMS und OGC WMTS / TMS eingeladen werden. Durch einen Klick auf **Datenquellen** kann zu einer Übersicht mit allen hinzugefügten Diensten navigiert werden. Diese ist wiederum in eine Liste mit allen Datenquellen sowie den freien Instanzen untergliedert. Mehr Informationen zu privaten und freien Instanzen finden sich unter *Layerset*.

Die Übersichtsseite **Datenquellen** bietet dem Nutzer folgende Funktionen:

- Datenquelle hinzufügen
- Datenquelle anzeigen
- Datenquelle aktualisieren
- Datenquelle löschen
- Datenquellen filtern



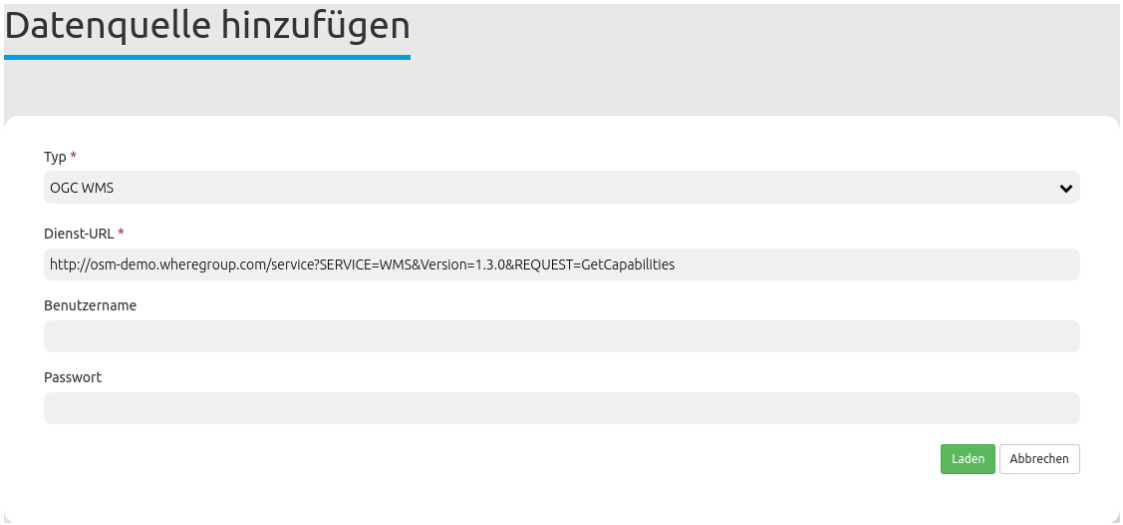
2.5.1 Laden von Datenquellen

Sie können OGC Web Map Services (WMS) und Web Map Tile Services (WMTS) in Ihre Anwendung laden. Mapbender unterstützt dabei die Versionen 1.0.0 und 1.3.0. Ein Dienst liefert ein XML, wenn das getCapabilities-Dokument angefordert wird. Diese Information wird von Mapbender ausgelesen. Der Client erhält alle notwendigen Informationen über den Dienst aus diesem XML.

Tip: Sofern möglich, sollten das Capabilities-Dokument zuerst in Ihrem Browser überprüfen, bevor Sie versuchen, den Dienst in Mapbender zu laden.

---

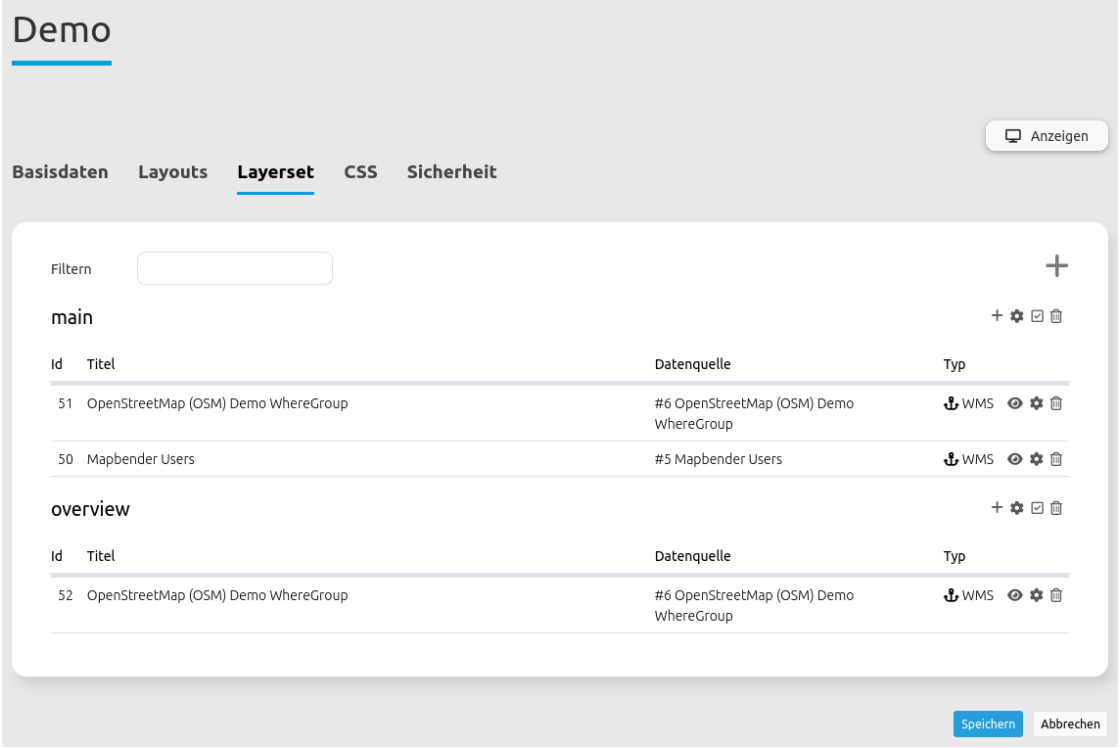
1. Um eine neue Datenquelle einzuladen, klicken Sie auf den Button **Datenquelle hinzufügen**.
2. Definieren Sie den **Typ** des Dienstes: *OGC WMS* oder *OGC WMTS / TMS*.
3. Geben Sie den Link zur getCapabilities URL in das Textfeld **Dienst-URL** ein.
4. Geben Sie Benutzername und Kennwort ein, sofern der Dienst eine Authentifizierung benötigt.
5. Klicken Sie **Laden**, um den Dienst in das Dienst-Repository zu laden.
6. Nach erfolgreicher Registrierung des Dienstes zeigt Mapbender eine Übersicht der Informationen an, die der Dienst geliefert hat.



2.5.2 Hinzufügen von Diensten zu Anwendungen

Nachdem ein Dienst in Mapbender geladen wurde, kann dieser zu einer oder mehreren Anwendungen hinzugefügt werden.

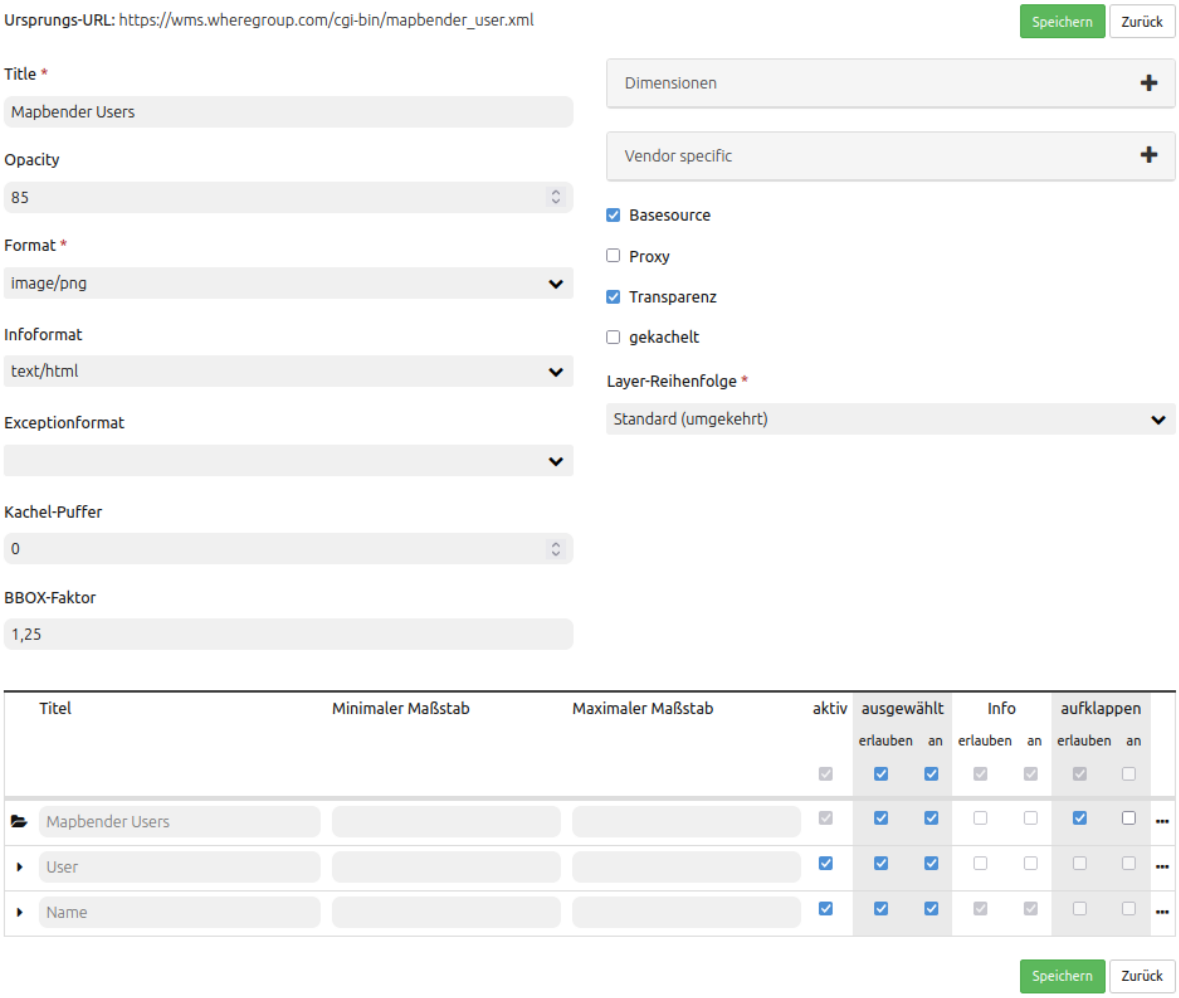
1. Navigieren Sie zunächst zur Seite **Anwendungen**. Klicken Sie dann auf bei der jeweiligen Anwendung und navigieren Sie zum Tab **Layersets**.
2. Im Bereich **Layersets** besteht die Möglichkeit, einzelne hochgeladene Dienste zu einer Anwendung hinzuzufügen. Klicken Sie hierfür zunächst auf + neben der Filterfunktion, um ein Layerset anzulegen. Alle Dienste in einer Anwendung müssen einem bestimmten Layerset zugeordnet sein. Vergeben Sie für dieses einen Titel (z.B. *main* für die Hauptkarte und *overview* für die Übersichtskarte).
3. Jetzt können Sie Dienste zum Layerset hinzufügen. Wählen Sie dafür + des jeweiligen Layersets.
4. Die Reihenfolge der Dienste kann über Drag & Drop verändert werden.



2.5.3 Konfiguration von Diensten

Sie können Dienste für Ihre Anwendung konfigurieren. Vielleicht möchten Sie sich nicht alle Ebenen anzeigen lassen, oder Sie möchten die Reihenfolge oder den Titel der Ebenen ändern, die Info-Abfrage für einzelne Ebenen verhindern oder den Minimal-/Maximalmaßstab ändern.

1. Wählen Sie **Anwendung** → → **Layerset** → **Instanz bearbeiten**, um eine Instanz zu konfigurieren.
2. Sie können nun Ihren Dienst konfigurieren.
3. Sie können die Reihenfolge der Layer über Drag & Drop ändern.



Dienstekonfiguration:

- **Titel**: Titel des Dienstes.
- **Opacity**: Durchlässigkeit (Deckkraft) des Dienstes in Prozent (0: transparent, 100: undurchsichtig).
- **Format**: Das Format für den getMap-Request.
- **Infoformat**: Das Format für getFeatureInfo-Requests (text/html für die Ausgabe als HTML wird empfohlen).
- **Exceptionformat**: Das Format für Fehlermeldungen.
- **Kachel-Puffer**: Dieser Parameter gilt für Dienste, die gekachelt angefordert werden und gibt an, ob weitere umgebende Kacheln abgerufen werden sollen. Damit sind diese bei einer Pan-Bewegung schon heruntergeladen und sichtbar. Je höher der Wert, desto mehr umgebende Kacheln werden abgerufen (Standard: 0).
- **BBOX-Faktor**: Dieser Parameter gilt für Dienste, die nicht gekachelt angefordert werden. Hier kann angegeben werden, wie groß das zurückgegebene Bild sein soll. Ein Wert größer 1 wird ein größeres Kartenbild anfordern. Standard: 1.25
- **BaseSource**: Der Dienst soll als BaseSource behandelt werden: Eine Aktivierung ist sinnvoll für vollflächige Hintergrundkarten, wie bspw. Straßenkarte und Satellit, bei denen eine gleichzeitige Anzeige wenig sinnvoll ist.

- **Proxy**: Bei Aktivierung wird der Dienst über Mapbender als Proxy angefordert. Der Browser fordert dann nicht direkt den Dienst an. Für passwortgeschützte Dienste sollte diese Option immer aktiviert sein, da ansonsten das Passwort für jeden Benutzer auslesbar ist.
- **Transparenz**: standardmäßig aktiviert, bei Deaktivierung wird der Dienst ohne transparenten Hintergrund angefordert (getMap-Request mit `Transparent=FALSE`).
- **gekachelt**: Dienst wird in Kacheln angefordert. Kann bei großer Karte sehr hilfreich sein, wenn der Dienst die Kartengröße nicht unterstützt (Standard: nicht gekachelt).
- **Layer-Reihenfolge**: gibt eine Lesrichtung für die im Dienst enthaltenen Layer vor. *Standard* (umgekehrt) und *QGIS-Style* (gleiche Reihenfolge) sind auswählbar.

Dimensionen:

Diese Funktion ist für WMS-Dienste mit einer zeitlichen Dimension von Relevanz. Weitere Informationen hierzu finden Sie unter *Dimensions Handler*.

Vendor Specific Parameter:

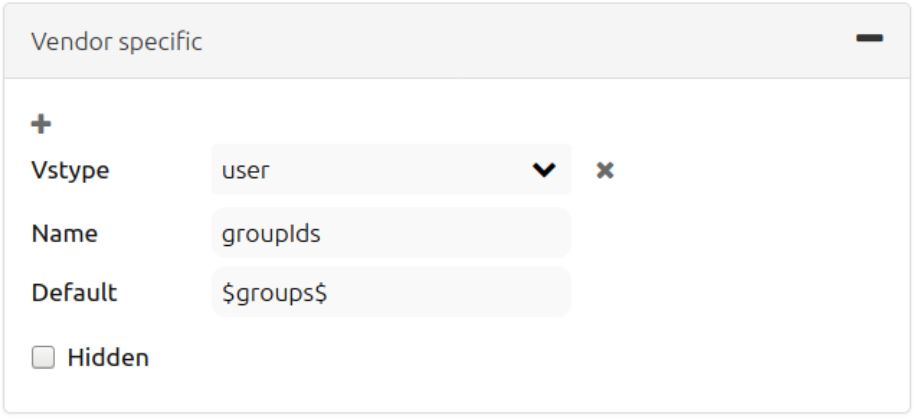
In einer Layerset-Instanz können Vendor Specific Parameter angegeben werden, die an den WMS Request angefügt werden. Die Umsetzung folgt den Angaben der multi-dimensionalen Daten in der WMS-Spezifikation.

In Mapbender können die Vendor Specific Parameter genutzt werden, um z.B. Benutzer und Gruppeninformation des angemeldeten Benutzers an die WMS Anfrage zu hängen. Es können auch feste Werte übermittelt werden.

Liste der möglichen Variablen:

- **User**: $email$, $groups$, id, $username$
- **Groups**: id, $title$, $description$

Das folgende Beispiel zeigt die Definition eines Parameters `group`, der als Inhalt die Gruppe des gerade in Mapbender angemeldeten Nutzers weitergibt.



- **Vstype**: Mapbender spezifische Variablen: simple, Gruppe (groups), User (users)
- **Name**: Parameter-Name im WMS Request
- **Default**: Standardwert
- **Hidden**: Falls der Wert gesetzt ist, werden die Anfragen serverseitig versendet, sodass die Parameter nicht direkt sichtbar sind. Funktioniert nur, sofern der Dienst als **Proxy** angefragt wird.

Momentan eignet sich das Element, um Benutzer und Gruppeninformationen weiterzugeben, z.B. für Benutzer die id und für Gruppen den Parameter $groups$.

Layerkonfiguration:

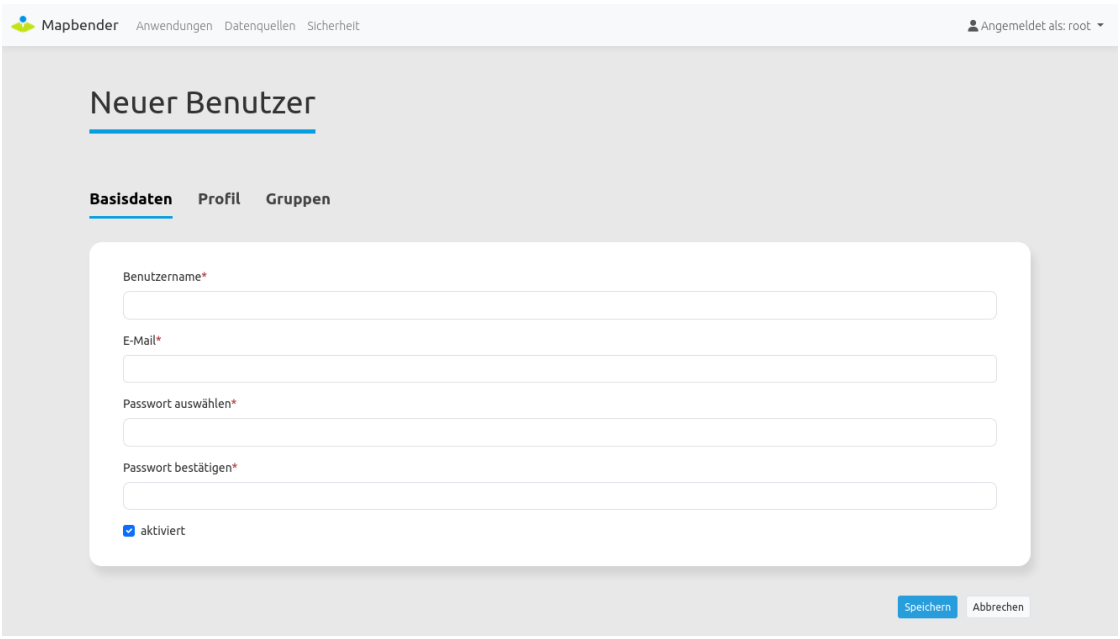
- **Titel**: Layertitel, wie er im Ebenenbaum angezeigt wird. Als Standard wird der Titel aus dem getCapabilities-Request genutzt.
- **Min./Max. Maßstab**: Maßstabsbereich, in dem der Layer angezeigt wird (z.B.: 1:100 - 1:1000)
- **aktiv**: aktiviert/deaktiviert einen Layer, sodass bei Deaktivierung in der Hierarchie untergeordnete ignoriert werden
- **Ausgewählt erlauben**: Layer wird angezeigt und ist auswählbar im Ebenenbaum
- **Ausgewählt an**: Layer ist bei Anwendungsstart aktiv
- **Info erlauben**: Infoabfrage wird für diesen Layer zugelassen
- **Info an**: Layer-Infoabfrage wird beim Start aktiviert
- **Aufklappen erlauben**: Erlaubt Aufklappen des Layers im Ebenenbaum
- **Aufklappen an**: Klappt Layer beim Start der Anwendung auf
- **Weitere Informationen (Drei-Punkte-Button): Öffnet einen Dialog mit weiteren Layer-Informationen**
  - **ID**: ID des Layers. Nützlich etwa, um URL-Parameter *zu kontrollieren*.
  - **Name**: Layername der Service Information (wird beim getMap-Request verwendet und ist nicht veränderbar)
  - **Style**: Wenn ein WMS mehr als einen Stil anbietet, können Sie einen anderen Stil als den Standard-Stil (default) wählen.

2.6 5. Benutzer- und Gruppenverwaltung

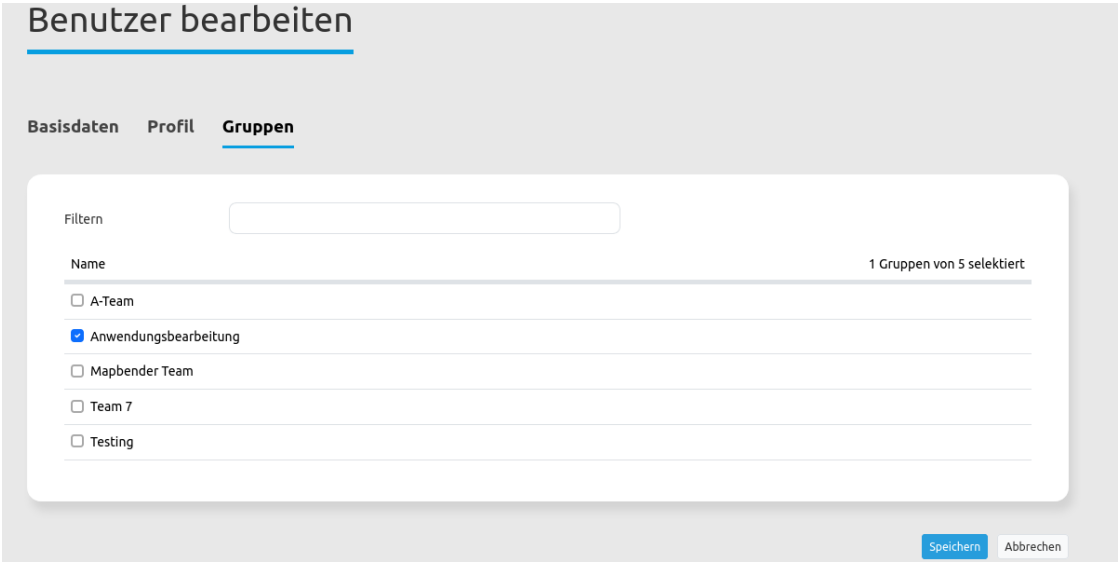
Der Zugriff auf eine Anwendung benötigt eine entsprechende Authentifizierung. Nur öffentliche Anwendungen können von allen genutzt werden. Benutzern oder Gruppen können Berechtigungen zugewiesen werden, um auf eine oder mehrere Anwendungen, Dienste oder mehr zuzugreifen.

2.6.1 Benutzer anlegen

1. Um einen Benutzer anzulegen, gehen Sie zu **Sicherheit Benutzer Neuen Benutzer hinzufügen**.
2. Wählen Sie einen Benutzernamen.
3. Geben Sie eine E-Mail-Adresse für den Benutzer an.
4. Wählen Sie ein Passwort für Ihren Benutzer aus und bestätigen Sie unter Passwort wiederholen.
5. Die Checkbox **aktiviert** gibt dem erstellten Account Zugriffsrechte auf das Mapbender-*Backend*.
6. Speichern Sie Ihren neuen Benutzer. Sie können alle Informationen auch nachträglich anpassen.



Weitere Angaben zum Benutzer können im Reiter **Profil** erfolgen. In den Reitern **Gruppen** und **Sicherheit** können dem Benutzer zusätzliche Parameter, z.B. die Zugehörigkeit zu einer Gruppe, zugewiesen werden.



2.6.2 Gruppen anlegen

1. Erzeugen Sie eine Gruppe über **Sicherheit** → **Gruppen** → **Neue Gruppe hinzufügen**.
2. Wählen Sie einen Namen und eine Beschreibung für Ihre Gruppe.
3. Weisen Sie der Gruppe über den Reiter **Benutzer** entsprechende Benutzer zu.
4. Speichern Sie Ihre neue Gruppe.

2.7 6. Rechteverwaltung

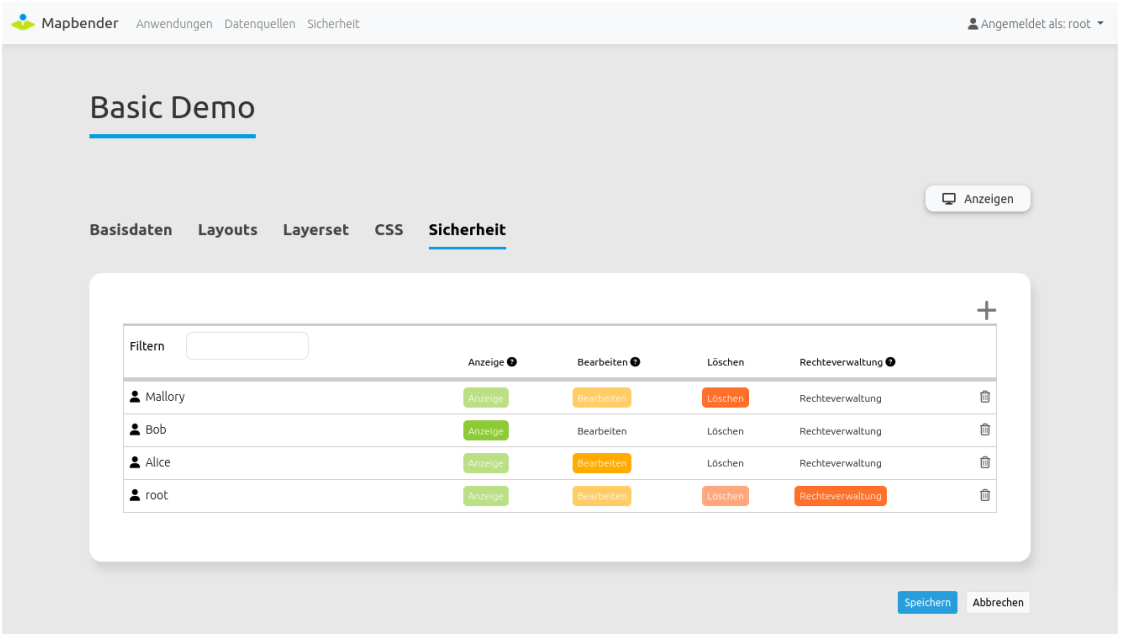
Mapbender bietet eine einfach zu bedienende Rechteverwaltung an, die im Backend implementiert ist.

- **Ansehen**: Anklicken, damit ein Benutzer die Berechtigung hat, sich Objekte (z.B. Anwendungen oder Datenquellen) ansehen zu können.
- **Erstellen**: Anklicken, damit ein Benutzer die Berechtigung hat, Objekte erstellen zu können.
- **Bearbeiten**: Anklicken, damit ein Benutzer die Berechtigung hat, Objekte bearbeiten zu können.
- **Löschen**: Anklicken, damit ein Benutzer die Berechtigung hat, Objekte löschen zu können.

Hint: Weisen Sie einem Benutzer globale Berechtigungen über **Sicherheit** → `Globale Berechtigungen` zu.

2.7.1 Zuweisen einer einzelnen Anwendung zu einem Benutzer/einer Gruppe

1. Bearbeiten Sie Ihre Anwendung über **Anwendungen** → .
2. Wählen Sie **Sicherheit**.
3. Veröffentlichen Sie Ihre Anwendung über die Auswahl des Benutzers **Öffentlicher Zugriff**. Alternativ wählen Sie **Öffentlichen Zugriff umschalten** in der Anwendungsübersicht. Ist diese Einstellung aktiviert, haben auch nicht angemeldete Nutzer einen Zugriff auf die Anwendung.
4. Alternativ zum öffentlichen Zugriff können Sie individuelle Zugriffsrechte für einen Benutzer oder eine Gruppe setzen:



Testen Sie die Konfiguration. Melden Sie sich oben rechts über **Abmelden** ab und melden sich unter einer neuen Benutzerbezeichnung an.

2.7.2 Zuweisen einzelner Elemente zu Benutzern/Gruppen

Standardmäßig stehen alle Elemente den Benutzern/Gruppen zur Verfügung, die Zugriff auf eine Anwendung haben. Der Zugriff kann darüber hinaus für einzelne Elemente noch genauer definiert werden, sodass diese nur bestimmten Benutzern/Gruppen zur Verfügung stehen.

1. Bearbeiten Sie Ihre Anwendung über **Anwendungen** .
2. Wählen Sie *Layouts*.
3. Jedes Element kann über **Zugriff auf das Element einschränken** eingeschränkt werden.
4. Wählen Sie **Zugriff auf das Element einschränken** zu dem Element, das nur ausgewählten Benutzern/Gruppen zur Verfügung stehen soll.
5. Weisen Sie dem Element Benutzern/Gruppen zu. Dadurch erhalten diese die Berechtigung **Anzeige** und können das Element in der Anwendung sehen.
6. Testen Sie die Konfiguration.

2.8 7. Anwendung beim Start positionieren

Sie können eine Anwendung beim Start positionieren. Dies kann über einen Punkt erfolgen. Beim Start können auch Texte zur Anzeige mitgegeben werden. Diese Funktionalität nutzt das Element Treffpunkt (MeetingPoint).

Sie können dabei einen oder mehrere Punkte (POIs) in der URL übergeben. Jeder Punkt verfügt dabei über die folgenden Parameter:

- **Punkt** (point): Koordinatenpaar, die Werte werden mit Komma getrennt (zwingend)
- **Beschriftung** (label): Beschriftung, die angezeigt werden soll (optional)
- **Maßstab** (scale): Maßstab, in dem der Punkt angezeigt werden soll (optional, Angabe ist nur bei der Anzeige eines Punktes sinnvoll)

Wenn Sie mehr als einen Punkt im Aufruf übergeben, zoomt die Karte auf 150% der POI-Boundingbox.

Format für die Übergabe eines einzelnen Punktes:

- ?poi[point]=363374,5621936&poi[label]=Hello World&poi[scale]=5000

2.9 Was kommt als Nächstes?

Dies waren nur die ersten Schritte auf dem Weg zur Nutzung von Mapbender. Es gibt noch eine Menge mehr zu entdecken:

- Besuchen Sie die offizielle Mapbender-Website,
- Finden Sie schnelle Antworten auf die wichtigsten Mapbender-bezogenen Fragen in der *FAQ - Häufig gestellte Fragen*,
- Werden Sie Mitglied der Mapbender-Community.

CHAPTER

THREE

ELEMENTE

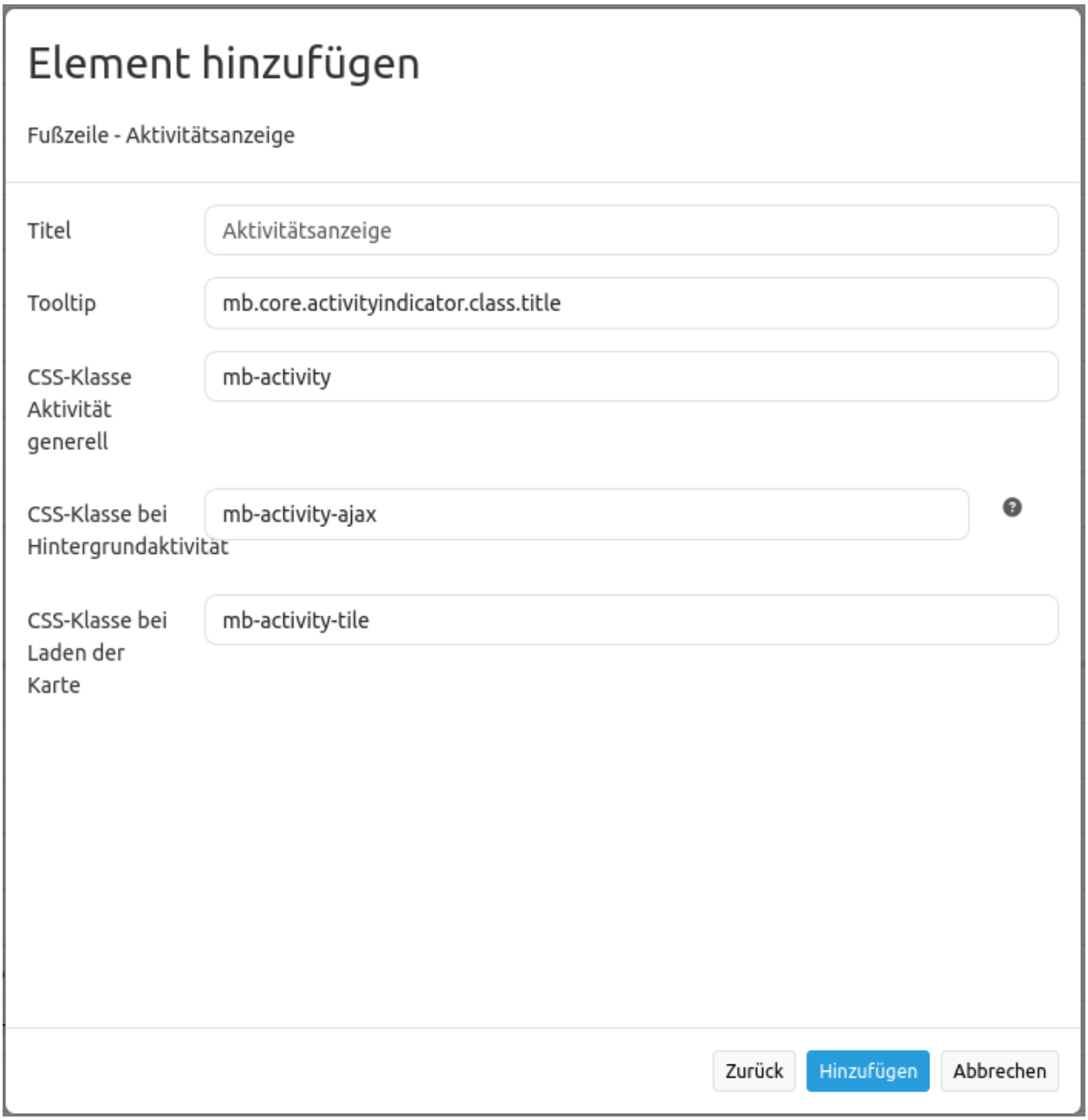
3.1 Basis-Elemente

3.1.1 Aktivitätsindikator (Activity Indicator)

Der Aktivitätsindikator (Sanduhr) ist ein einfaches Modul, das Aktivitäten anzeigt (Ajax-Aufrufe und Kartenaufrufe). In der voreingestellten Konfiguration wird ein Schrift-Symbol verwendet.

Der Aktivitätsindikator kann in Sidepane, oberer Werkzeugleiste oder Fußzeile implementiert werden. Standardmäßig sieht er folgendermaßen aus:

Konfiguration



- **Title:** Titel des Elements
- **Tooltip:** Der hier eingegebene Text wird angezeigt, wenn der Mauszeiger längere Zeit über dem Element verweilt.
- **Activity class:** CSS Klasse, die Aktivitäten anzeigt (Ajax oder Kacheln)
- **Ajax activity class:** CSS Klasse, die Ajax-Aktivitäten anzeigt
- **Tile activiy class:** CSS Klasse, die Kartenaufrufe anzeigt

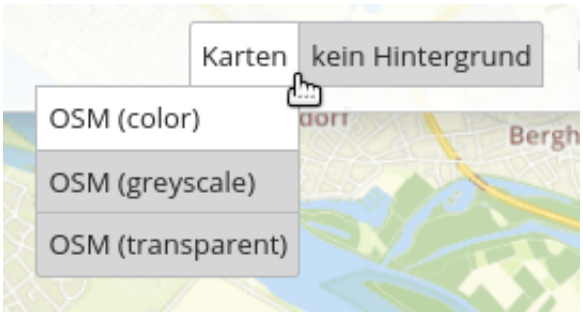
YAML-Definition

Diese Vorlage kann genutzt werden, um das Element in einer YAML-Anwendung einzubinden.

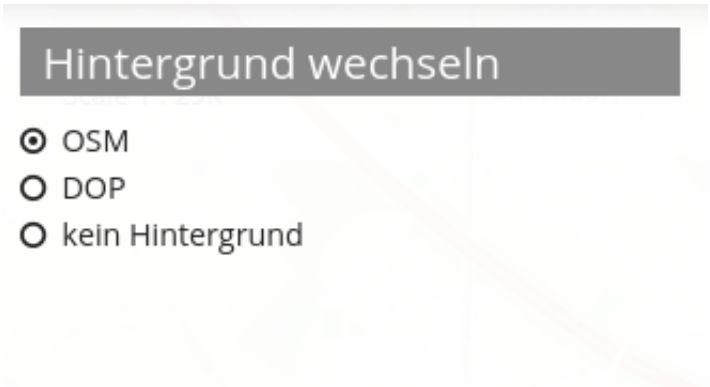
```
activityClass: mb-activity          # CSS Klasse, die Aktivitäten anzeigt (Ajax oder Kacheln)
ajaxActivityClass: mb-activity-ajax # CSS Klasse, die Ajax-Aktivitäten anzeigt
tileActivityClass: mb-activity-tile # CSS Klasse, die Kartenaufrufe anzeigt
```

3.1.2 Hintergrund wechseln

"Hintergrund wechseln" erstellt Buttons mit Themen, um zuvor für diesen Zweck konfigurierte Dienste im Kartenbereich sichtbar zu schalten. Auf diese Weise kann etwa zwischen verschiedenen Hintergrundkarten gewechselt werden. Dabei ist immer nur ein Thema aktiv. Zusätzlich können Gruppen für die Hintergrund-Themen definiert werden. Alle Definitionen einer Gruppe werden als Dropdown-Liste angezeigt, wobei der Gruppenname als übergeordneter Eintrag erscheint.

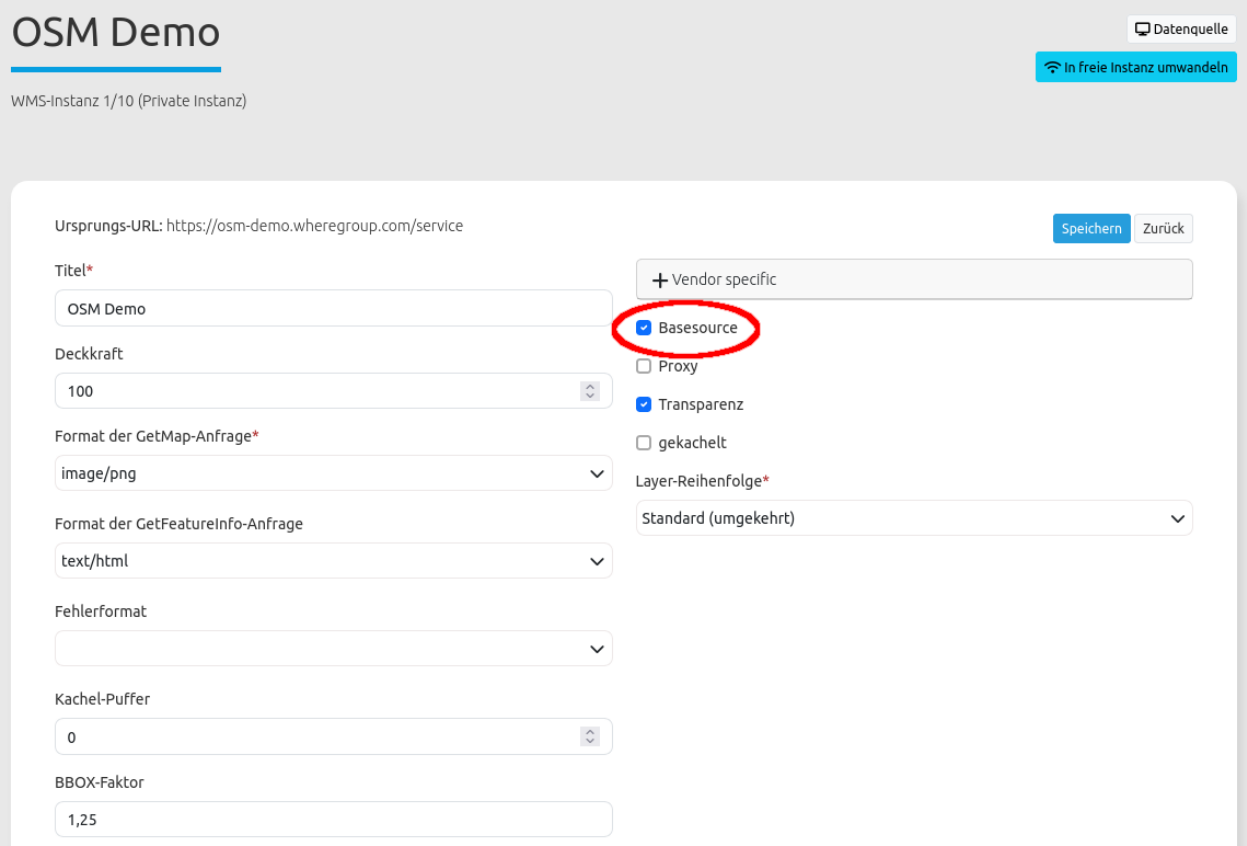


Hint: Das Element kann auch in der Sidepane eingebunden werden. Beim Hinzufügen des Hintergrundwechslers in der Sidepane ist keine Definition von Gruppen möglich.



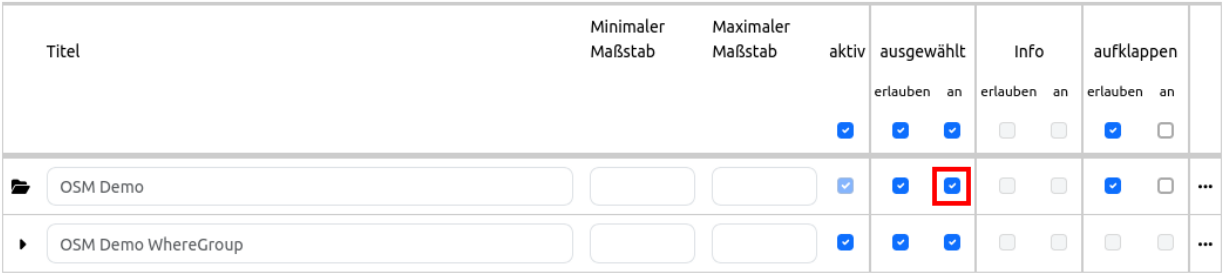
Konfiguration

Vorbereitung: Zur Definition einer Hintergrundkarte mit dem Element müssen zuvor im *Backend* (Reiter: *Layerset*) mindestens zwei Dienste konfiguriert werden. Um sie verwenden zu können, ist in den Layerset-Einstellungen eines Dienstes das Anhaken der Checkbox “Basesource” erforderlich.

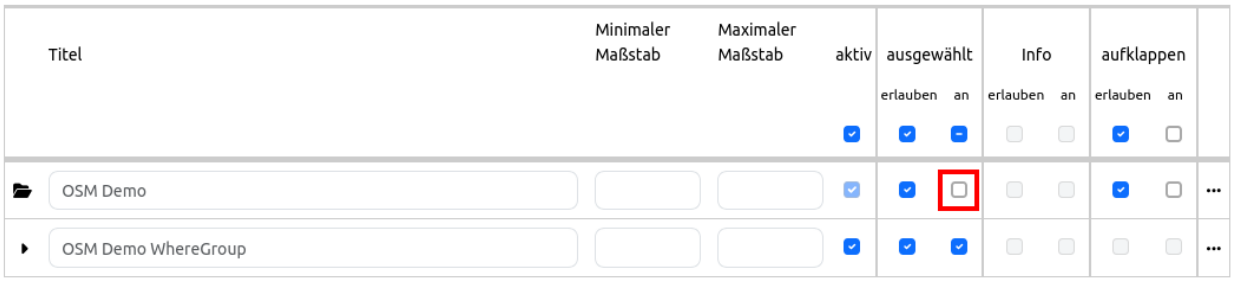


Note: Beachten Sie, dass beim Anwendungsstart die Themen aktiviert werden, bei denen der root-Layer aktiv ist.

Konfiguration aktiver ausgewählter root-Layer - Thema ist aktiv beim Start:



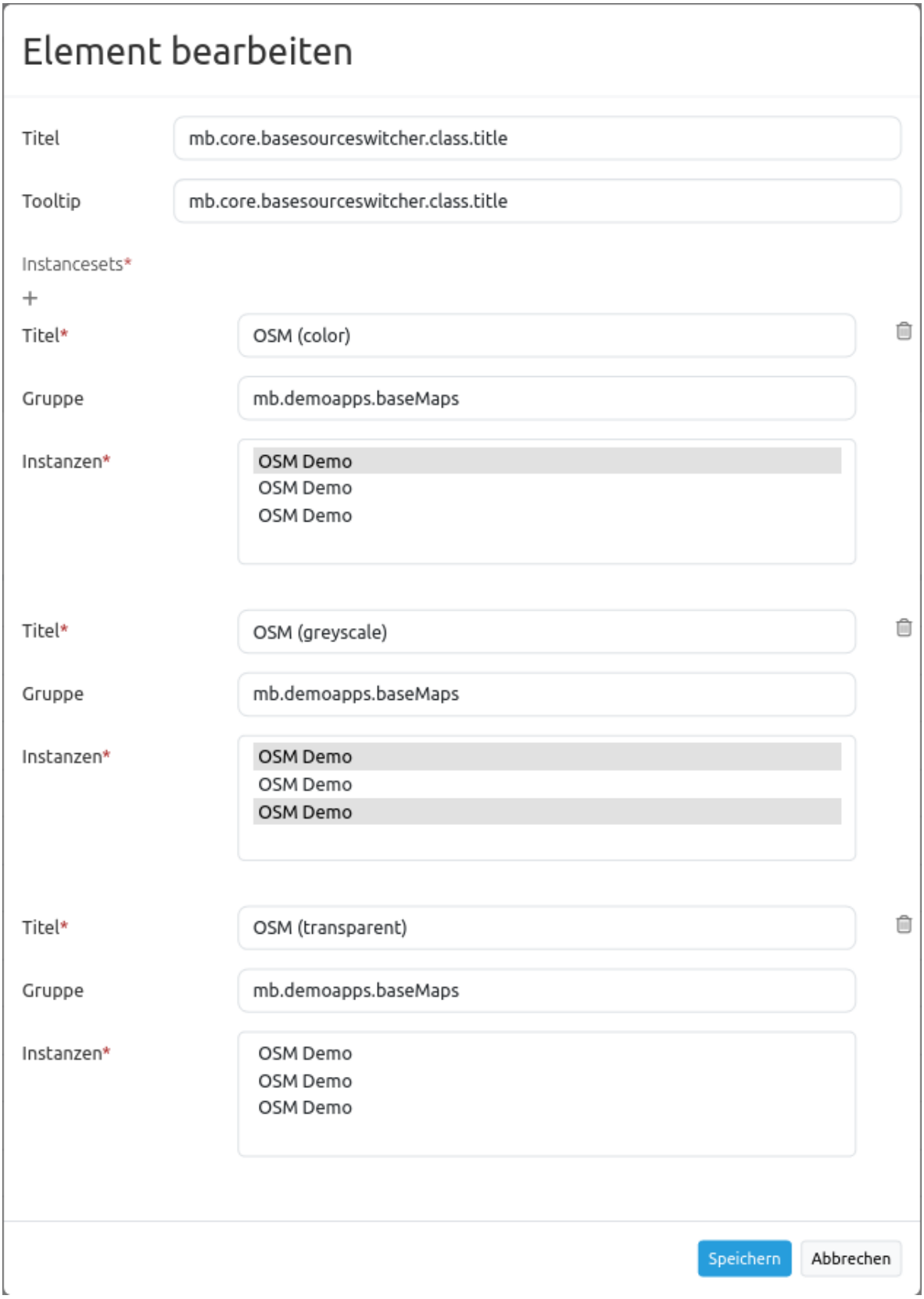
Konfiguration nicht aktivierter ausgewählter root-Layer - Thema ist beim Start nicht aktiv:



Die Konfiguration des Elements geschieht in zwei Schritten:

1. Erzeugen eines Elements zum Wechseln der vordefinierten Themen (Titel, Tooltip; bei Einbindung in Kartenbereich: Position)
2. Hinzufügen von Themen mit einer oder mehreren Quellen und optionaler Definition einer Gruppe

- **Titel:** Titel des Elements.
- **Tooltip:** Text, der erscheint, wenn der Mauszeiger längere Zeit über dem Hintergrundwechsler gehalten wird.
- **Instancesets:** Es können eine oder mehrere Themengruppen definiert werden. Diese verweisen auf eine Auswahl an Instanzen und sind mit einem Titel und einem Gruppennamen (optional) zu versehen.



Das Konfigurationsbeispiel zeigt, dass entweder ein, kein oder mehrere Einträge pro Instanceset gewählt

werden können. Durch eine Group-Angabe lassen sich Gruppen bilden. Weitere Instancesets können über den + Button hinzugefügt werden. Ein Instanceset lässt sich mit Drag & Drop verschieben.

- **Titel**: Name der Hintergrundkarte.
- **Gruppe**: Optionale Zuweisung zu einer Themengruppe.
- **Position:** Positionierung (nur bei Nutzung im Kartenbereich). Möglichkeiten: 'Oben links', 'Unten links', 'Oben rechts', 'Unten rechts'
- **Instanzen**: Auswahl der Instanzen für die Hintergrundkarte.

YAML-Definition

Diese Vorlage kann genutzt werden, um das Element in einer YAML-Anwendung einzubinden.

```
title:                                          # Titel
tooltip:                                        # Text des Tooltips
target: map                                     # ID des Kartenelements
anchor: 'right-bottom'                          # Positionierung nur bei Nutzung im Kartenbereich
↪(Standard: right-bottom) - Optionen: 'left-top', 'right-top', 'left-bottom', 'right-bottom'
sourcesets:                                     # Liste der Sourcesets.
    - { title: sourcesetname, group: groupname,
        sources: [sourceId]}                    # sourceset: Titel,
                                                # group: (optional) Gruppenname der Gruppen der
↪Sourcesets über "group name"
                                                # sources Liste der Sources
    - { title: sourcesetname, group: groupname,
        sources: [sourceId]}
```

3.1.3 Koordinatenanzeige (Coordinates Display)

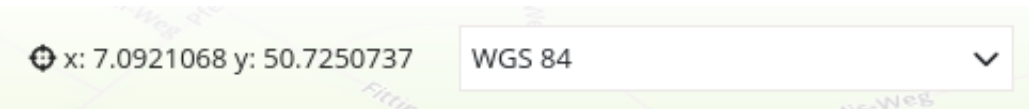
Die Koordinatenanzeige zeigt die Mausposition in den Kartenkoordinaten an. Die Koordinaten sind abhängig vom eingestellten räumlichen Referenzsystem, welches im *SRS Auswahl (SRS Selector)* geändert werden kann.

So sieht die Koordinatenanzeige für unterschiedliche Koordinatensysteme aus:

ETRS89 / UTM zone 32N (EPSG:25832):

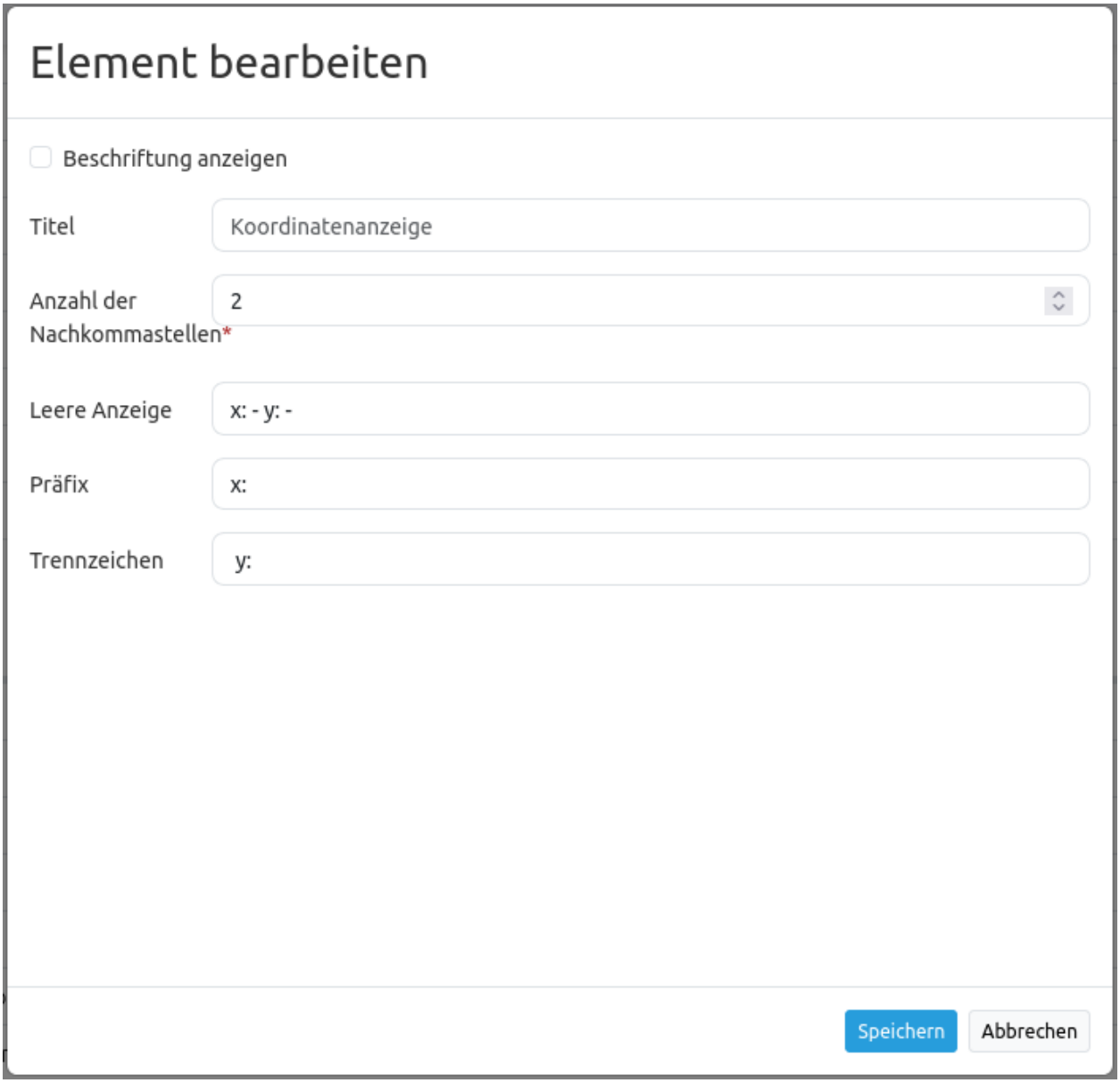
DHDN / 3-degree Gauss-Kruger zone 2 (EPSG:31466):

WGS 84 (EPSG:4326):



WGS 84 / Pseudo-Mercator (EPSG:3857):

Konfiguration



- **Beschriftung anzeigen:** Schaltet die Beschriftung an/aus. Die Beschriftung richtet sich nach dem Title.
- **Titel:** Titel des Elements. Dieser wird links neben den Koordinaten angezeigt, wenn “Beschriftung anzeigen” aktiviert ist.
- **Anzahl der Nachkommastellen:** Nachkommastellen der angezeigten Koordinaten.

- **Leere Anzeige:** Angezeigter Text, wenn sich die Maus nicht in der Karte befindet (Standard: 'x= - y= -').
- **Präfix:** Präfix vor der X-Koordinate (Standard: 'x= ').
- **Trennzeichen:** Separator nach X- und vor Y-Koordinate (Standard: ' y= ').
- **Position:** Positionierung (Pflichtfeld, nur bei Nutzung im Kartenbereich).

YAML-Definition

Diese Vorlage kann genutzt werden, um das Element in einer YAML-Anwendung einzubinden.

```yaml
numDigits: 2                    # die Anzahl der Nachkommastellen, die jede Koordinate haben soll
target: map                     # ID des Kartenelements
label: true                     # false/true, um den Button zu beschriften (Standard: true).
empty: 'x= - y= -'              # zeigt diesen Text, wenn die Maus sich nicht in der Karte befindet.
prefix: 'x= '                   # zeigt ein Präfix vor der X-Koordinate.
separator: ' y= '               # zeigt einen Separator vor der Y-Koordinate.
anchor: 'right-bottom'          # Positionierung nur bei Nutzung im Kartenbereich (Standard: right-
↪bottom) - Optionen: 'left-top', 'right-top', 'left-bottom', 'right-bottom'
```

CSS-Styling

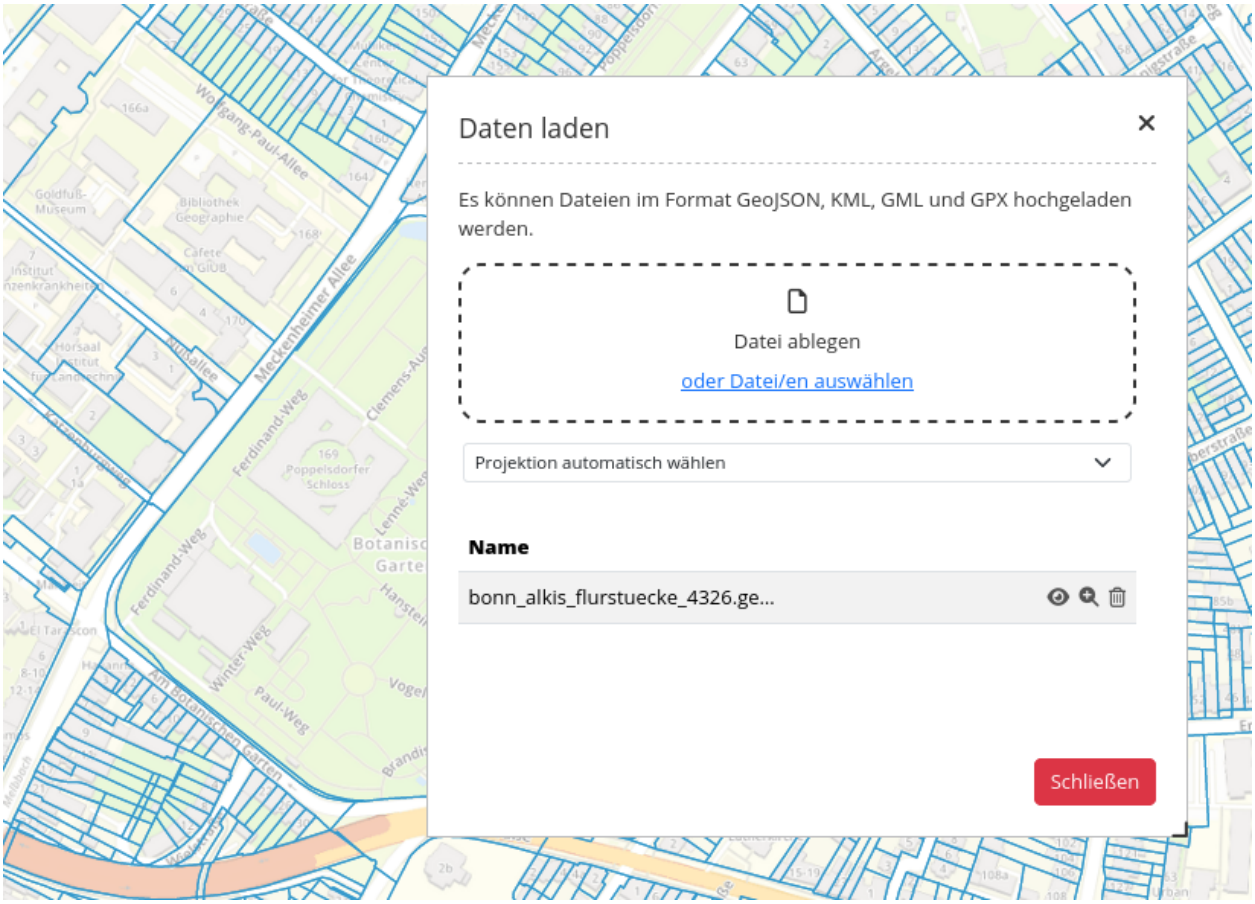
Das Element kann über den folgenden CSS-Style angepasst werden, beispielsweise um die Breite zu vergrößern.

```css
.mb-element-coordsdisplay {
    width: 500px;
}
```

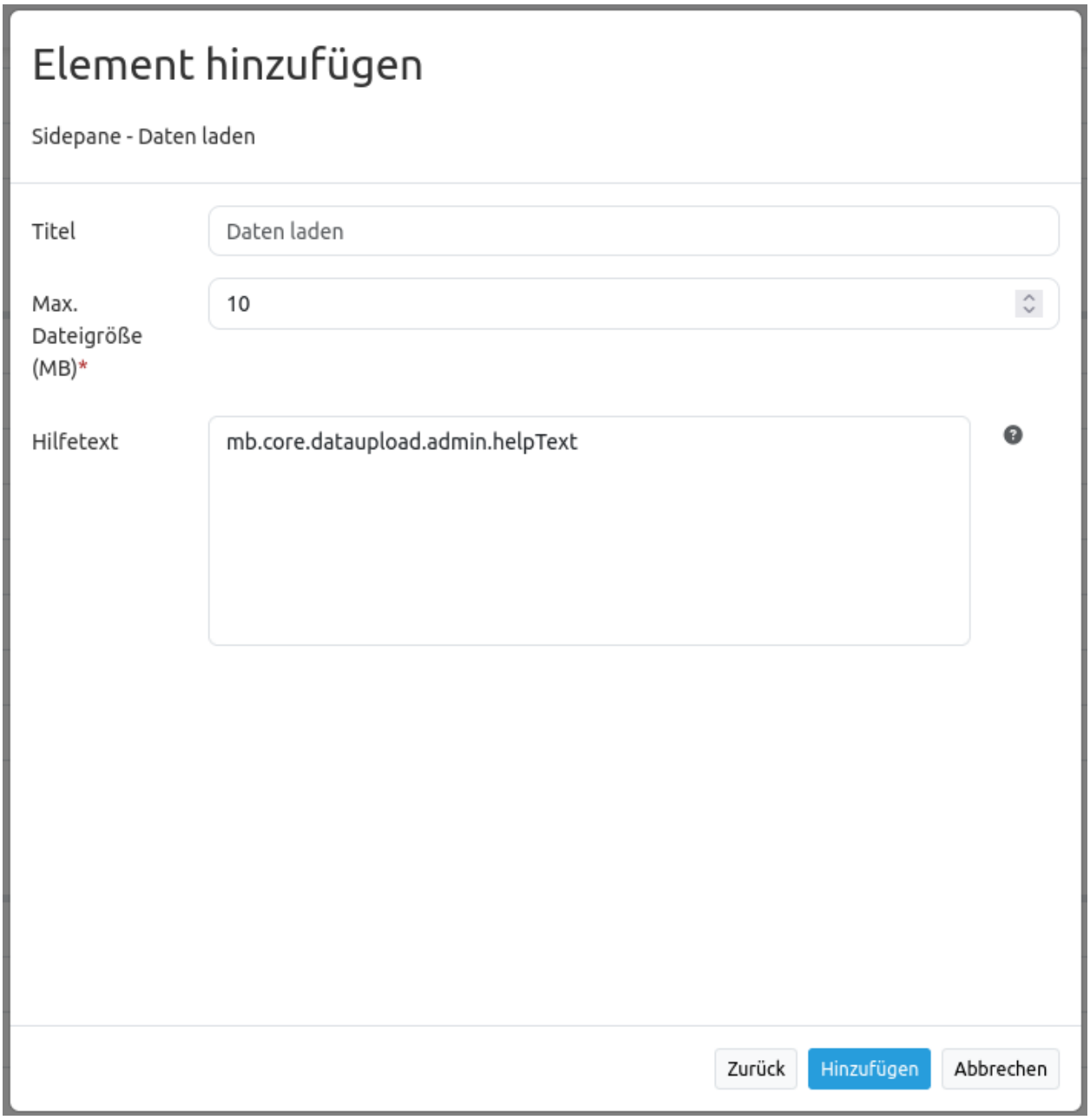
3.1.4 Daten laden

Mit **Daten laden** können Punkte, Linien, Polygone und Multipolygone aus Geodaten in den Formaten GeoJSON-, KML-, GML- und GPX geladen werden. Sie können eine oder mehrere Dateien per Drag-and-Drop oder über die Dateiauswahl hochladen. Für eine passende Projektion wählen Sie entweder ein eigenes Koordinatenreferenzsystem oder nutzen das Feature *Projektion automatisch bestimmen*.

Hint: Die Dateigröße ist begrenzt. In der Elementkonfiguration kann eine maximale Größe definiert werden.


Daten laden
Es können Dateien im Format GeoJSON, KML, GML und GPX hochgeladen werden.
Datei ablegen
oder Datei/en auswählen
Projektion automatisch wählen
Name
bonn_alkis_flurstuecke_4326.ge...
Schließen

Konfiguration



- **Titel**: Titel des Elements, erscheint in der Kopfzeile des Elements.
- **Max. Dateigröße (MB)**: Maximal erlaubte Dateigröße in Megabyte. Vorgabewert: 10 MB.
- **Hilfetext**: Frei definierbarer Hilfetext, der über dem Uploadfeld angezeigt wird.

Nachdem **Daten laden** dem Backend hinzugefügt wurde, wird es in Mapbender an der entsprechenden Position angezeigt. Anschließend können neue Dateien sowohl über Drag-and-Drop als auch über den 'Datei wählen'-Button hinzugefügt werden. Nachdem eine Datei hochgeladen wurde, werden die Objekte auf der Karte angezeigt und in der Sidepane gelistet. Über die Liste können Sie die Objektsichtbarkeit einstellen, auf das Objekt zoomen oder es löschen.

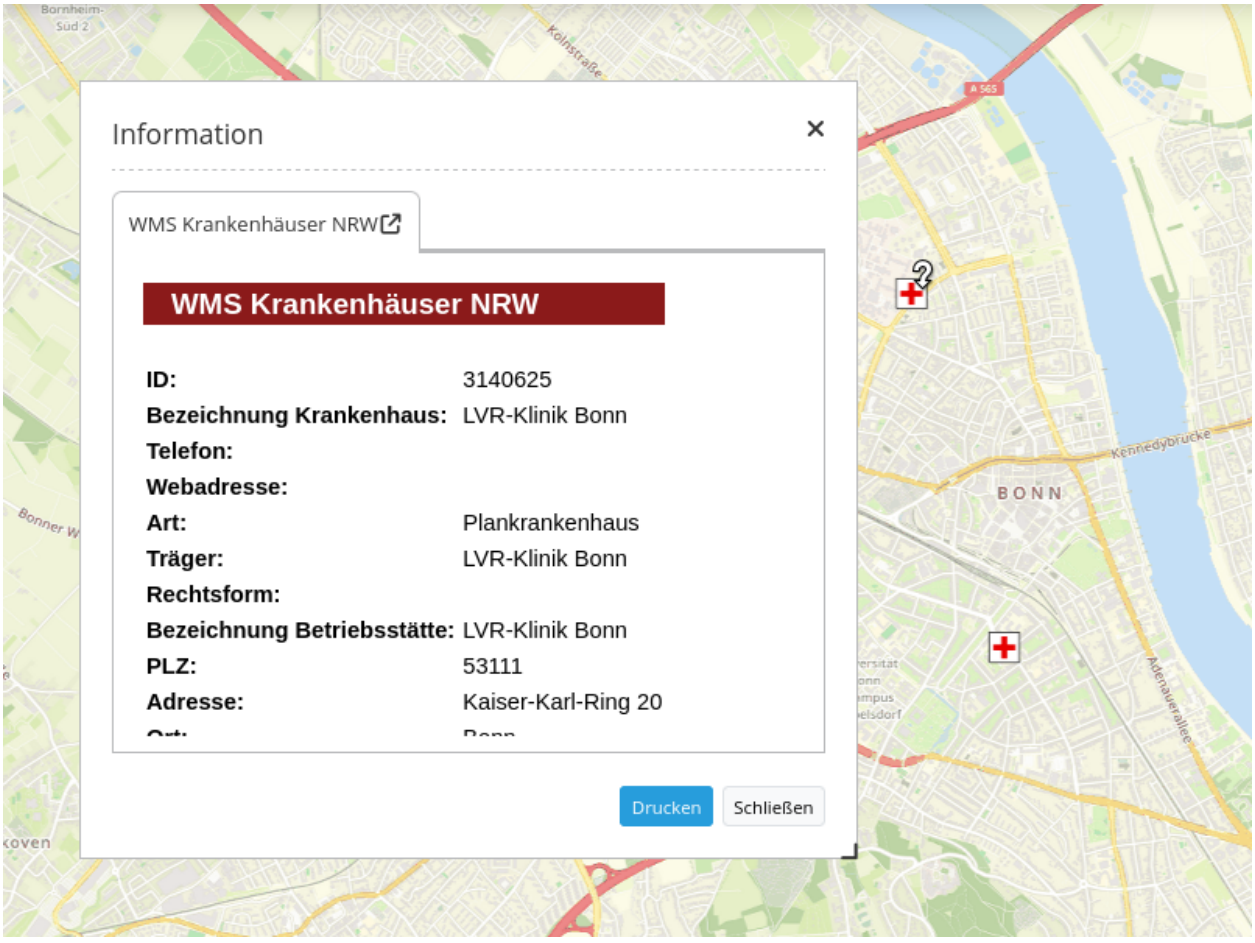
YAML-Definition

Diese Vorlage kann genutzt werden, um das Element in einer YAML-Anwendung einzubinden.

```
dataupload:
   class: Mapbender\CoreBundle\Element\DataUpload      # Element class name
   target: map                                         # Element target
   maxFileSize: 10                                     # Maximum file size (Megabyte)
   helpText: mb.core.dataupload.admin.helpText         # Text that appears as help dialog
```

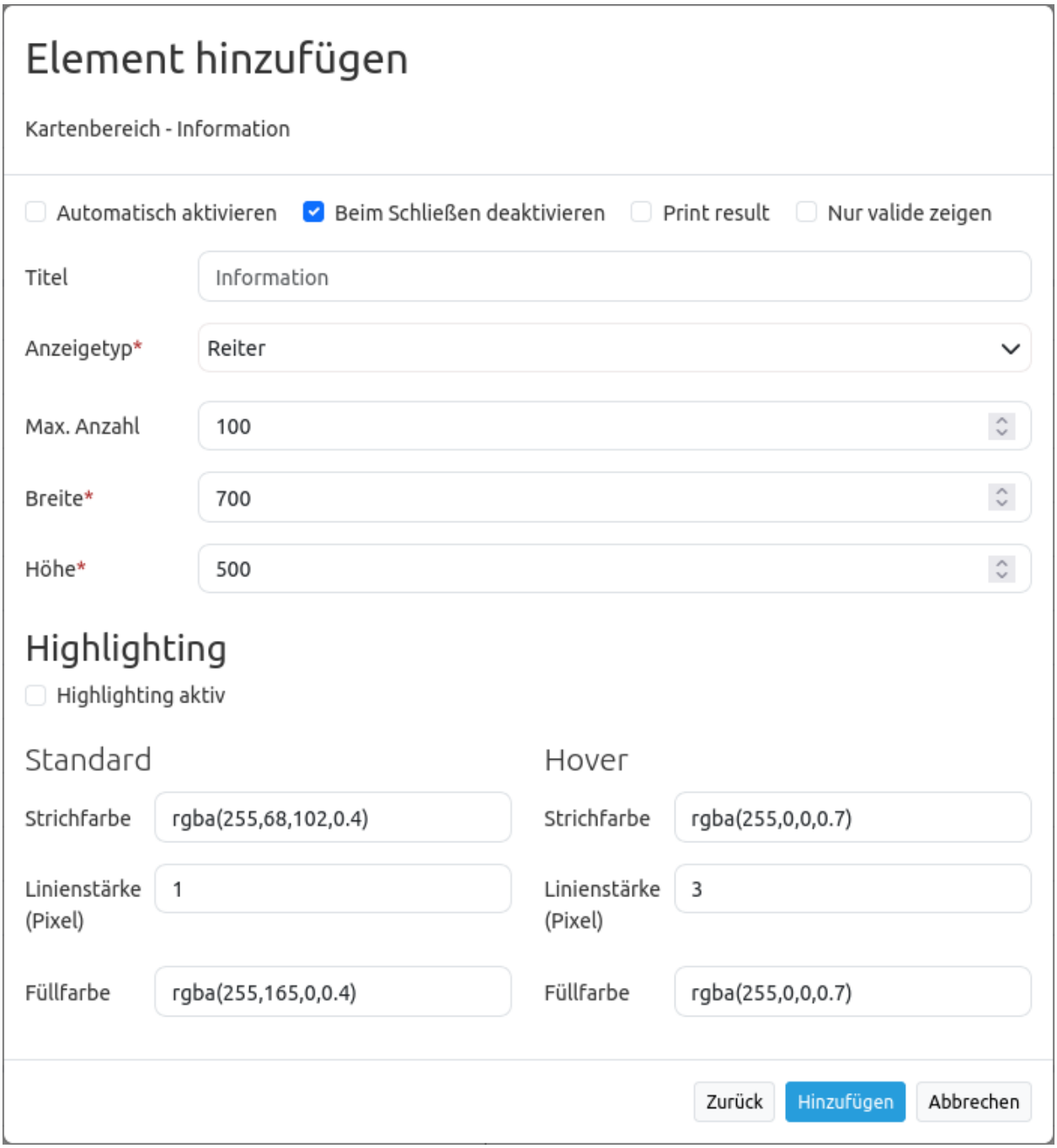
3.1.5 Information (FeatureInfo)

FeatureInfo bietet die Möglichkeit, mit einem Mausklick auf einen Layer Informationen über Features zu erhalten. Das Element funktioniert mit unterstützten WMS-Diensten. Es kann in die Sidepane oder in einen *Button* integriert werden.



Der Dienst Krankenhäuser NRW des Ministeriums für Gesundheit, Emanzipation, Pflege und Alter NRW dient zur Veranschaulichung.

Konfiguration



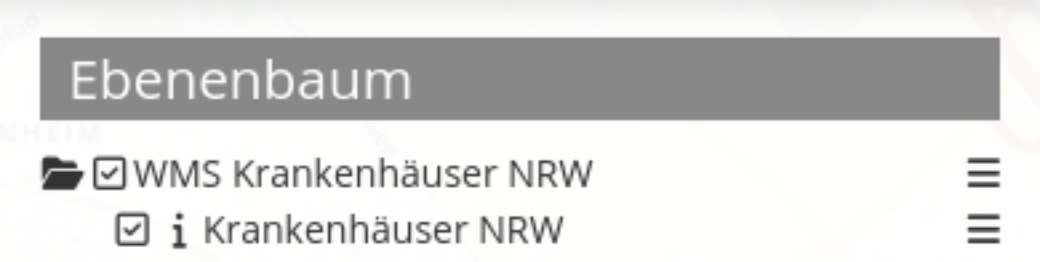
- **Automatisch aktivieren:** Schaltet ein/aus, ob das Informationsfenster beim Start der Anwendung automatisch geöffnet werden soll (Standard: false).
- **Beim Schließen deaktivieren:** Steuert, ob das FeatureInfo beim Schließen des Ergebnisfensters deaktiviert wird oder nicht (Standard: true).
- **Print result:** Anzeige eines Links, über den die abgefragten Daten ausgedruckt werden können (Standard: false).
- **Nur valide zeigen:** Anzeige von ausschließlich validen WMS (Standard: false).

- **Titel:** Titel des Elements. Dieser wird in der Layouts Liste angezeigt und ermöglicht, mehrere Button-Elemente voneinander zu unterscheiden. Der Titel wird außerdem neben dem Button angezeigt, wenn "Beschriftung anzeigen" aktiviert ist.
- **Anzeigetyp:** Anzeige verschiedener Dienstabfragen im Informationsfenster, entweder in verschiedenen Reitern oder als Akkordeon (Standard: tabs).
- **Max. Anzahl:** Maximale Anzahl an Treffern/Ergebnissen, die angezeigt werden soll.
- **Breite/Höhe:** Größe des Dialogfeldes (Breite und Höhe in Pixel).
- **Highlighting aktiv:** Aktivierung des FeatureInfo Highlightings.
- **Strichfarbe:** Umrandungsfarbe der ausgewählten Objekte.
- **Linienstärke (Pixel):** Setzt die Breite der Umrandungslinie in Pixeln.
- **Füllfarbe:** Füllfarbe der ausgewählten Objekte.
- **Hover-Strichfarbe:** Umrandungsfarbe der ausgewählten Objekte beim Hovern (Verweilen des Mauszeigers auf dem Objekt).
- **Hover-Linienstärke (Pixel):** Setzt die Breite der Umrandungslinie in Pixeln beim Hovern.
- **Hover-Füllfarbe:** Füllfarbe der ausgewählten Objekte beim Hovern (Verweilen des Mauszeigers auf dem Objekt).

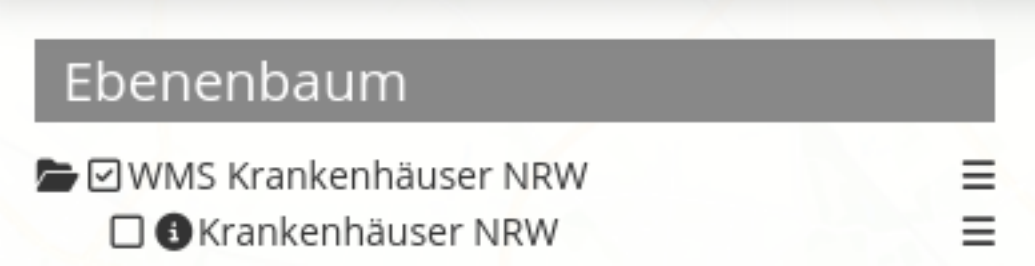
Einstellungen im Ebenenbaum

Layer ist sichtbar und FeatureInfo-Abfrage für den Layer ist aktiviert:

Layer ist sichtbar und FeatureInfo-Abfrage für den Layer ist deaktiviert:



Layer ist nicht sichtbar; es erfolgt keine FeatureInfo-Abfrage, auch wenn diese aktiviert ist:



Ausdruck der Resultate

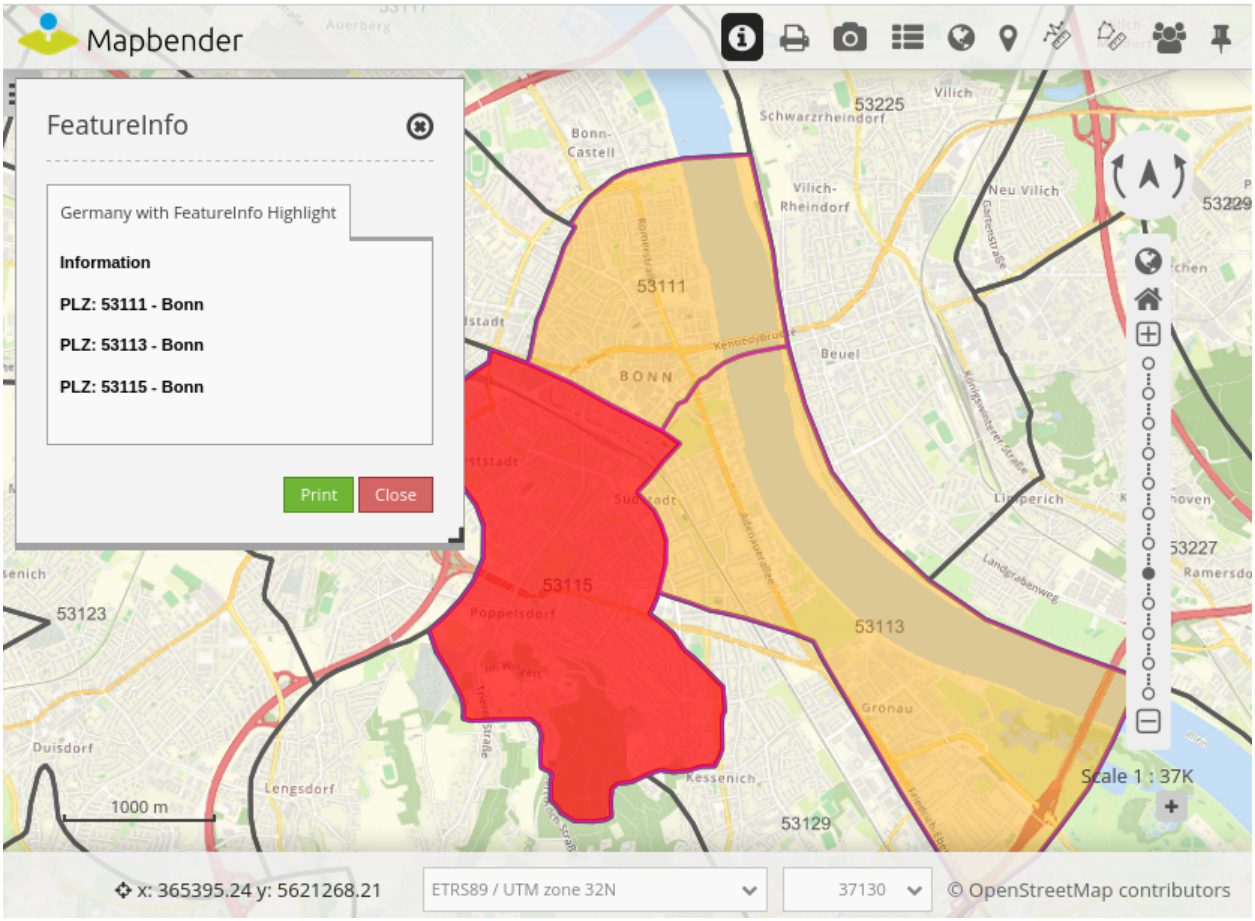
Über die **Print Result**-Checkbox kann der Informationsdruck aktivierbar, sodass im Anschluss ein Drucken-Button im FeatureInfo-Dialog sichtbar ist. Der Druckprozess selbst geschieht über den Druckdialog des Webbrowsers.

Um alle Bilder und Hintergrundfarben im Ausdruck zu erhalten, sollten Sie die Druckeinstellungen des Webbrowsers beachten: In Firefox heißt die Option “Hintergrund drucken” und wird im Druckoptionendialog angeschaltet, in Chrome-basierten Browsern nennt sich die Option “Hintergrundgrafiken”. Die übermittelten Schriften können bei einem Ausdruck in PDF je nach Viewer unterschiedlich gut funktionieren. Des Weiteren modifizieren die meisten Browser Webseiten etwas vor dem Druck, damit nicht so viel Tinte/Toner verbraucht wird.

FeatureInfo Highlighting

Einzelne Geometrien eines WMS können über die Infoabfrage in der Karte farblich hervorgehoben werden. Dies ist beispielsweise bei der Arbeit mit umfangreichen WMS hilfreich, da so einzelne Geometrien leichter zugeordnet werden können.

Eine Infoabfrage mit aktiviertem FeatureInfo Highlighting könnte beispielsweise folgendermaßen aussehen:



In der Abbildung wurden mehrere Geometrien in der Karte ausgewählt (PLZ: 53111, 53113 und 53115). Der FeatureInfo Dialog zeigt die Informationen zu diesen Objekten an. Die Fläche mit der PLZ 53115 wird durch Hovering rot in der Karte markiert.

Das FeatureInfo Highlighting kann im FeatureInfo-Element im Kartenbereich aktiviert werden. Dort sind auch Grund- und Hoverfarbe auswählbar.

Darüber hinaus muss die HTML-Ausgabe der Infoabfrage angepasst werden. Hierfür ist es notwendig, dass die Geometrieabfrage versteckt als WKT über ein HTML-div erfolgt. Zusätzlich muss der EPSG-Code übergeben werden und eine eindeutige ID in dem HTML-div vorliegen (siehe Konfigurationsbeispiel unten). Mapbender wertet nach korrekter Konfiguration diese Informationen aus und stellt die Geometrien in der Karte dar. Beim Mouse-Over auf den Treffern des Infofensters wird die dazugehörige Geometrie farblich entsprechend hervorgehoben. Anpassungen können für MapServer, QGIS Server, GeoServer erfolgen.

Die notwendige Anpassung wird hier am Beispiel von MapServer gezeigt. In der DATA-Angabe wird zusätzlich die Geometrie als WKT ausgegeben. Außerdem wird das FeatureInfo-Template angepasst. Wird nun ein WMS über GetFeatureInfo abgefragt, werden die entsprechenden Flächen in der Karte hervorgehoben. Sofern gewünscht kann auch via data-label eine Beschriftung definiert werden.

```
DATA "geom from (Select *, ST_AsText(geom) as geom_wkt from plz) as foo USING UNIQUE gid USING SRID 4326
↪"

<div class="geometryElement" id="[gid]" data-geometry="[geom_wkt]" data-srid="EPSG:4326" data-label=
↪"[beschriftung]" >
<table>
```

(continues on next page)

(continued from previous page)

```
      ...
<table>
</div>
```

YAML-Definition

Diese Vorlage kann genutzt werden, um das Element in einer YAML-Anwendung einzubinden.

```
title: FeatureInfo              # Titel des Elements
tooltip: Feature Info           # Text des Tooltips
type: dialog                    # Standard, Pflichtfeld: dialog
target: map                     # ID des Kartenelements
autoActivate: false             # true, wenn die Infoabfrage beim Start der Anwendung geöffnet wird
↪(Standard: false)
deactivateOnClose: true         # true/false, um die Funktion nach dem Schließen des Ergebnisfensters zu
↪deaktivieren (Standard: true)
onlyValid: false                # Korrekte HTML Ausgabe erfordern (Standard: false).
printResult: false              # Anzeige eines Links, über den die Infoabfrage ausgedruckt werden kann
↪(Standard: false)
displayType: tabs               # tabs/accordion (Standard: tabs)
width: 700                      # Breite des Dialogs in Pixel (Standard: 700)
height: 500                     # Höhe des Dialog in Pixel (Standard: 500)
maxCount: 100                       # Maximale Anzahl an Treffern/Ergebnissen, die angezeigt werden soll
highlighting: false             # FeatureInfo Highlighting (Standard: false)
fillColorDefault: '#ffff00'     # Füllfarbe zur Hervorhebung ausgewählter Objekte
strokeColorDefault: '#ff00ff'   # Umrandungsfarbe zur Hervorhebung ausgewählter Objekte
opacityDefault: 25              # Opazität (%) der ausgewählten Objekte
strokeWidthDefault: 3           # Breite der Umrandungslinie der ausgewählten Objekte
fillColorHover: '#00ffff'       # Füllfarbe zur Hervorhebung beim Hovern über ausgewählte Objekte
strokeColorHover: '#0000ff'     # Umrandungsfarbe zur Hervorhebung beim Hovern über ausgewählte Objekte
opacityHover: 50                # Opazität (%) der ausgewählten Objekte beim Hovern
strokeWidthHover: 5             # Breite der Umrandungslinie der ausgewählten Objekte beim Hovern
```

3.1.6 Ebenenbaum (Layertree)

Funktionen

- Anzeige von Layern
- Anzeige von Layergruppen
- Anzeige thematischer Layergruppen
- Aktivieren und Deaktivieren von Layern
- Aktivieren und Deaktivieren der Abfrage von Layern
- Verändern der Layerreihenfolge
- Zoom zum Kartenausschnitt eines Layers
- Abfrage der Metadaten eines Layers

- Zeitliche Dimension eines Layers kontrollieren

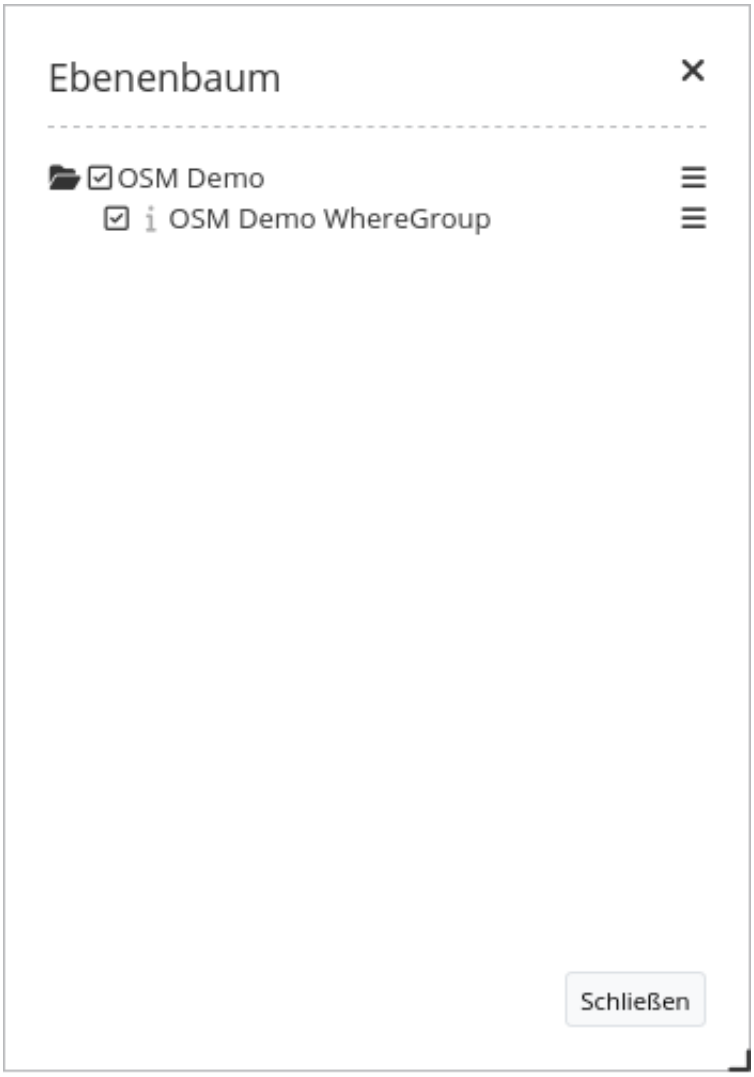


Fig. 1: Einfacher Ebenenbaum im Dialogfenster zeigt Dienste mit Ebenen.

Zur Konfiguration des Ebenenbaums gibt es verschiedene Verknüpfungspunkte zu anderen Elementen, die beachtet werden müssen:

- *Layerset*
- *Karte (Map)*
- *Datenquellen (Sources)*

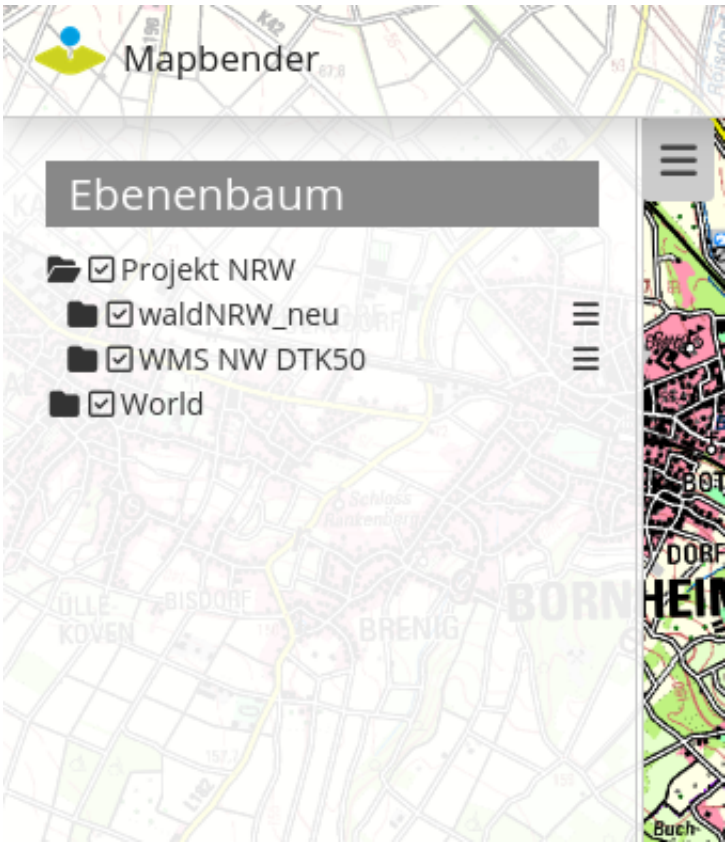


Fig. 2: Thematischer Ebenenbaum in der Sidepane

Konfiguration

Allgemeine Konfiguration

Um Layersets im Ebenenbaum nutzen zu können, sind verschiedene Anpassungen notwendig. Diese betreffen die:

1. Einrichtung unterschiedlicher Layersets
2. Einrichtung in der Karte zur Anzeige der Layersets
3. Einrichtung des Ebenenbaums selbst

Über den Reiter **Layerset** können Layersets definiert und Datenquellen in eine Anwendung eingebunden werden.

Die Instanzen sind die Referenzen auf die einzelnen WMS-Dienste. Über + kann ein neues Layerset hinzugefügt werden. In dieses lassen sich im Anschluss neue Layer über das Hinzufügen von im *Backend* registrierten Instanzen einbinden. Das Layerset "overview" wird im Beispiel für die Anzeige der Übersichtskarte verwendet. Eine Dokumentation, wie die Dienste korrekt registriert und eingebunden werden, findet sich unter *Datenquellen (Sources)* und *Layerset*.

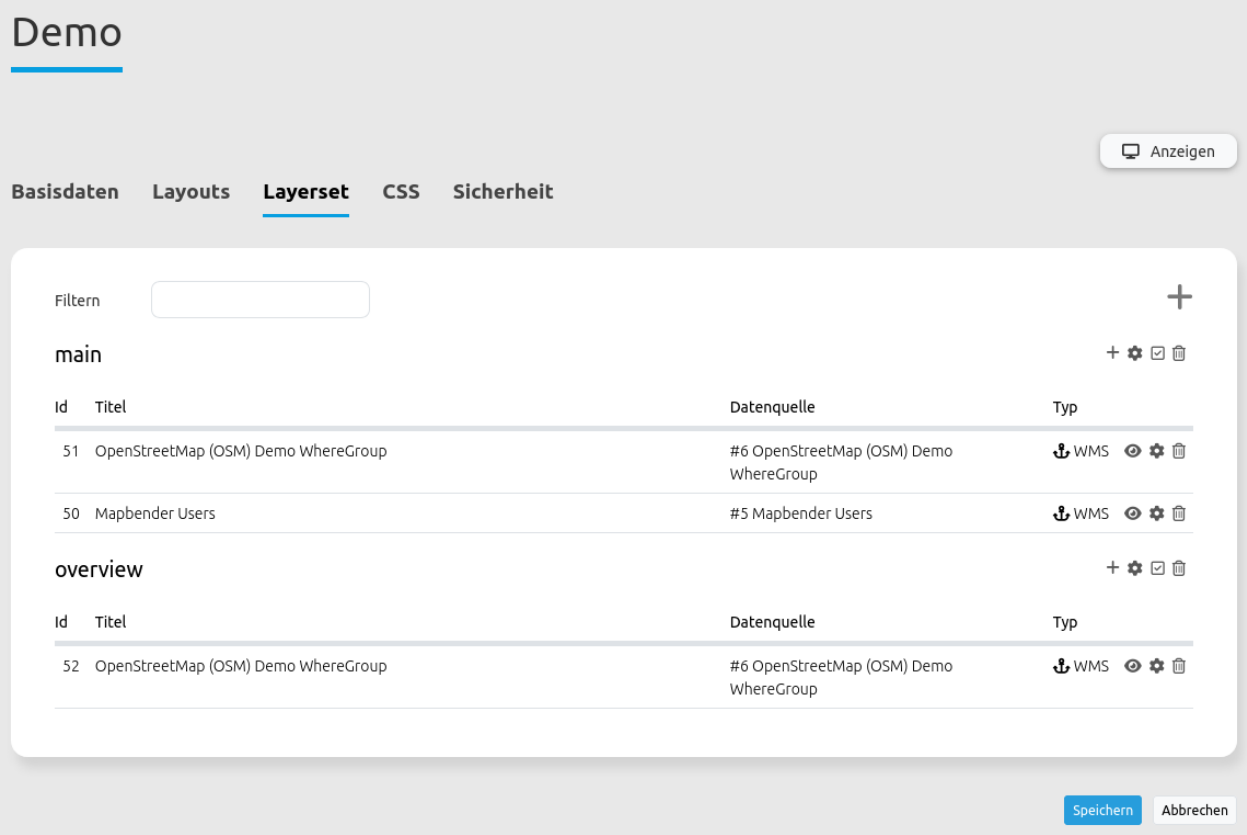


Fig. 3: Einrichtung von Layersets für die Einbindung in den Ebenenbaum.

Damit die neu eingebundenen Layersets auch in der Anwendung erscheinen, müssen diese im *Kartenelement* angegeben werden. Durch das Aktivieren der gewünschten Layersets wird definiert, ob diese in der Kartenansicht verwendet werden. Auch wird hierüber die Reihenfolge definiert, in der die Layersets im

Ebenenbaum und in der Karte erscheinen: Die Einträge in der Auflistung können per Drag & Drop verschoben werden.

Workflow einfacher Ebenenbaum

Im Folgenden gehen wir eine beispielhafte Konfiguration eines Ebenenbaums mit Basisfunktionen in drei Schritten durch:

1. Einrichtung eines Layersets
2. Einrichtung in der Karte zur Anzeige des Layersets
3. Einrichtung des Ebenenbaums

Einrichtung eines Layersets

Im Beispiel ist ein **Layerset** mit einer Instanz definiert:

- **Layerset World:**
  - Instanz OSM Demodienst

Die registrierte Instanz des OSM-Demodienstes wird bei der Installation von Mapbender bereits mitgeliefert. Diese muss nur noch über den Anwendungsreiter **Layerset** in ein Layerset eingebunden werden. Im Beispiel wurde das Layerset "World" genutzt. Bei Schwierigkeiten bei der Einbindung kann die Dokumentation der *Layerset* weiterhelfen.

Einrichtung in der Karte zur Anzeige des Layersets

Als Nächstes erfolgt die Einrichtung des *Kartenelements* zur Anzeige des Layersets in der **Karte**. Dazu wechseln wir in den Anwendungsreiter **Layouts** und bearbeiten dann das Kartenelement im Kartenbereich. Wichtig ist, dass im Bereich Layersets das Layerset "World" aktiviert wird, damit es in der Anwendung angezeigt wird.

Bei Fragen zur weiteren Konfiguration der Karte kann die Dokumentation des *Kartenelements* weiterhelfen.

Einrichtung des Ebenenbaums

Zuletzt erfolgt die Einrichtung des **Ebenenbaums** selbst. Die im Kartenelement angegebenen Instanzen können im Ebenenbaum präziser definiert werden. Wenn eine vordefinierte Anwendung kopiert wurde, sollte der Ebenenbaum bereits funktionieren. Beim erneuten Einbinden des Elements in die Anwendung kann dieses mit den Standardeinstellungen genutzt werden, ohne dass weitere Anpassungen zwingend notwendig sind.

Für das Verstehen der Funktionen und das eigene Erstellen eines Ebenenbaums bearbeiten wir nun das Ebenenbaum-Element im Kartenbereich.

Über die Checkbox **BaseSources anzeigen** werden Instanzen, die als Basesource in die Anwendung geladen wurden, auch im Ebenenbaum angezeigt.

Wenn **Automatisches Öffnen** aktiv ist, wird der Ebenenbaum beim Anwendungsstart geöffnet. Der **Titel** des Elements ermöglicht die Unterscheidung mehrerer eingebundener Ebenenbäume.

Über **Menü** kann eine Auswahl von Funktionen aktiviert werden, die dann im Ebenenbaum zur Verfügung gestellt werden.

- **Ebenen entfernen**: Entfernen von Layern aus dem Ebenenbaum erlauben
- **Deckkraft**: Deckkraft eines einzelnen Layers verändern
- **Auf Ebene zoomen**: Layer zentriert in seiner vollen Ausdehnung anzeigen

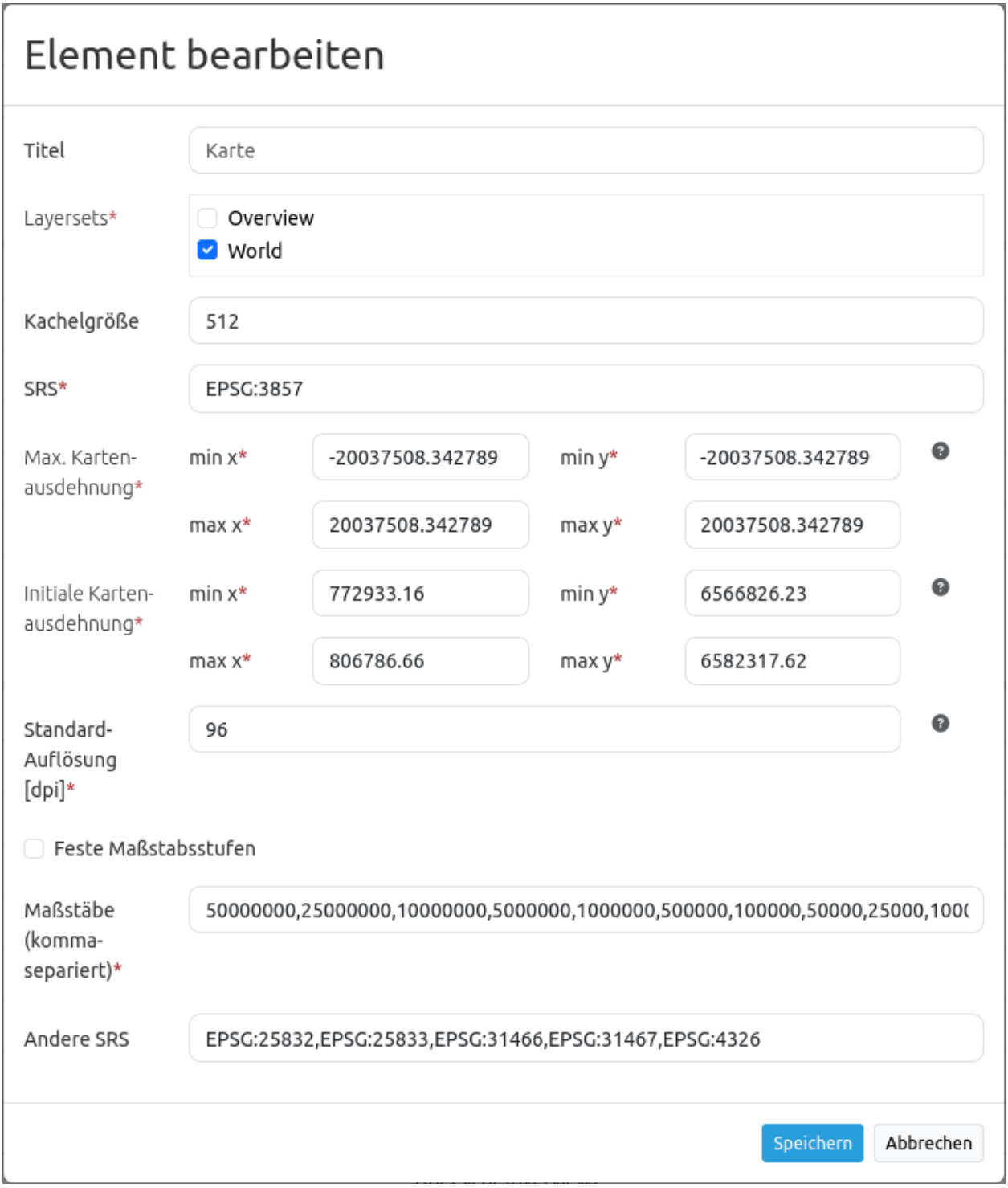


Fig. 4: Einrichtung der Layersets im Kartenelement.

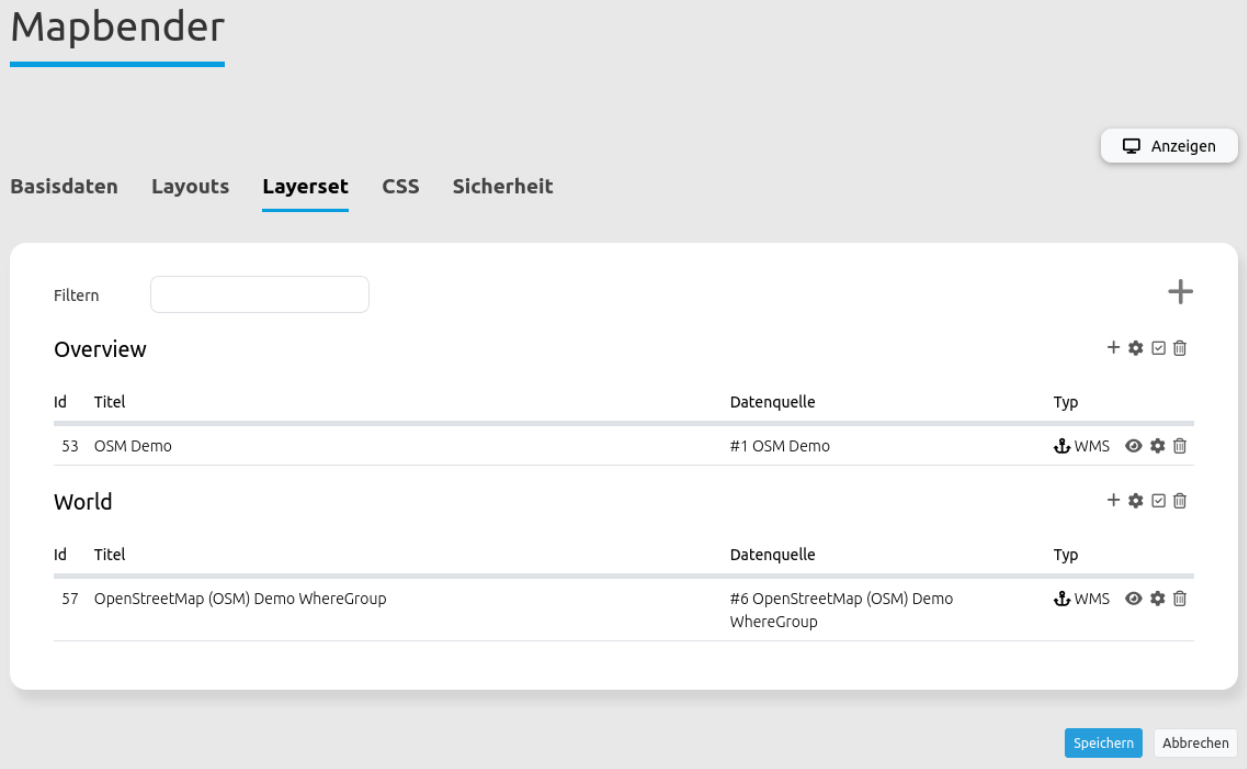


Fig. 5: Einrichtung eines Layersets für die Einbindung in den Ebenenbaum.

- **Metadaten**: Metadaten eines Layers anzeigen
- **Dimension**: Dimension eines Layers kontrollieren. Mehr Informationen unter *Dimensions-Handler*.
- **Stil auswählen**: Erlaubt die Möglichkeit den Stil in der Anwendung zu wechseln, wenn ein Layer mehrere Stile anbietet.

Die einzelnen Funktionen können durch Klicken auf die Schaltfläche aktiviert werden. Alle aktiven Funktionen werden farbig hinterlegt. In der Anwendung erscheint im Ebenenbaum rechts neben jedem Layer ein Kontextmenü. Durch einen Klick auf das Menü klappt ein Fenster auf und die zuvor aktivierten Funktionen können genutzt werden. Durch einen Klick auf den x-Button oben rechts kann das Menü wieder geschlossen werden.

Die Checkbox **Info ausblenden** ermöglicht das Ausblenden des "i"-Symbols; die Einstellungen zur Informationsabfrage sind unabhängig von den Einstellungen in Layerset oder Dienst. Das Symbol zeigt an, ob die Informationsabfrage aktiviert ist: Wenn es nicht umrandet ist, ist die Informationsabfrage nicht aktiv. Durch einen Klick wird die dunkle Umrandung aktiviert, sodass bei einer Informationsabfrage Layer-Informationen abgefragt werden können.

Die Checkbox **Thematischer Layer** aktiviert die thematische Einteilung von Layern. Dadurch werden die Layerset-Namen mit Ordnersymbol und Checkbox im Ebenenbaum angezeigt. Darüber hinaus kann definiert werden, ob ein Ordner beim Start geöffnet dargestellt werden soll. Durch einen Klick auf das Ordnersymbol des Ebenenbaums neben der Instanz können die eingebundenen Layer angezeigt werden. Alle Layer, die vorher bei dem Einbinden in das Layerset aktiviert wurden, erscheinen nun in der Liste.

Die Checkbox **Erlaubt sortieren im Layerbaum** steuert, ob ein freies Umsortieren von Layern per Drag & Drop im Ebenenbaum erfolgen darf.

Die Checkbox neben einem Layernamen ermöglicht das An- und Ausschalten eines Layers. Ist die Checkbox aktiviert, erscheint der Layer in der Karte. Es werden jedoch weiterhin die im Layerset definierten

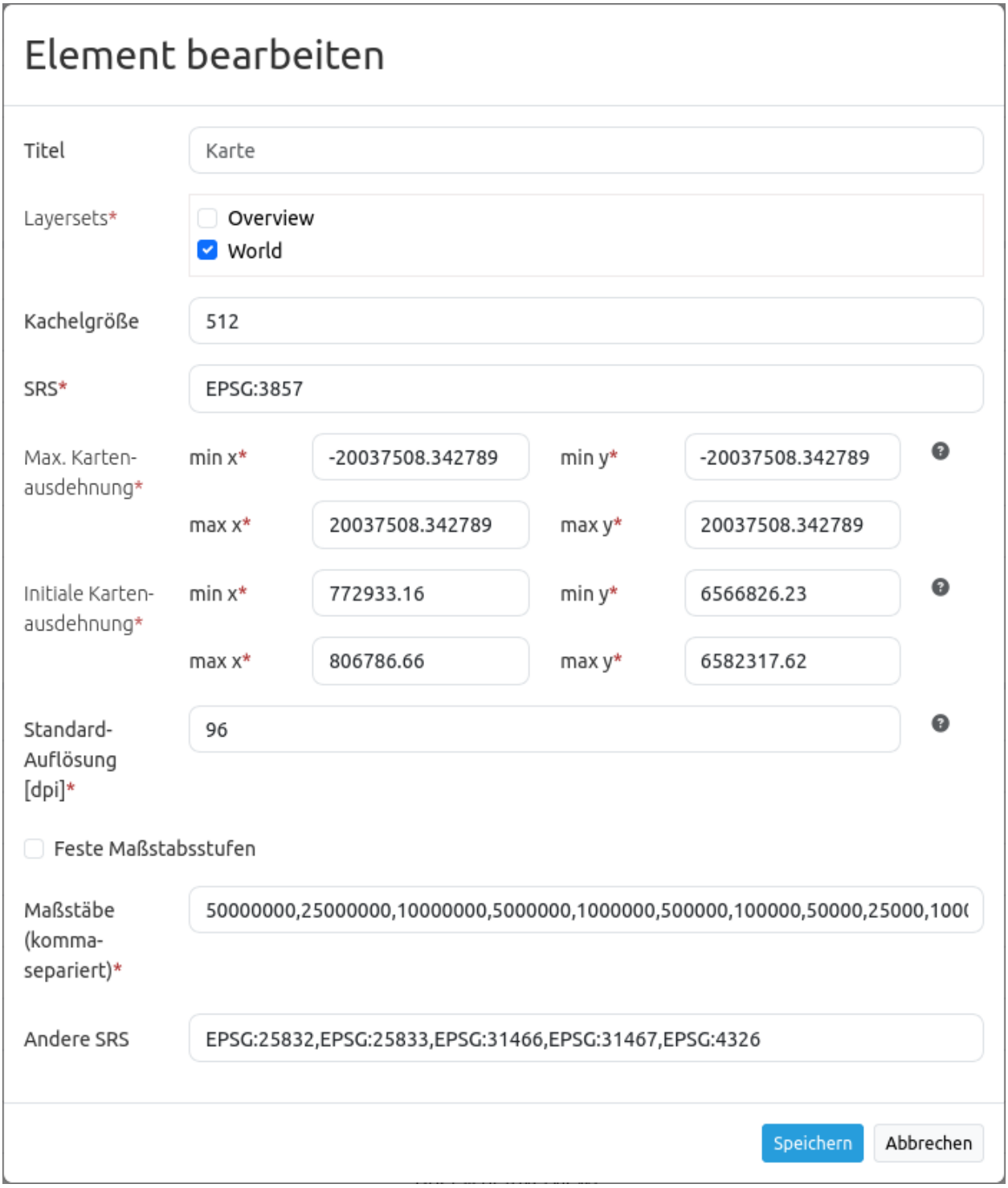


Fig. 6: Einrichtung im Kartenelement zur Anzeige des Layersets.

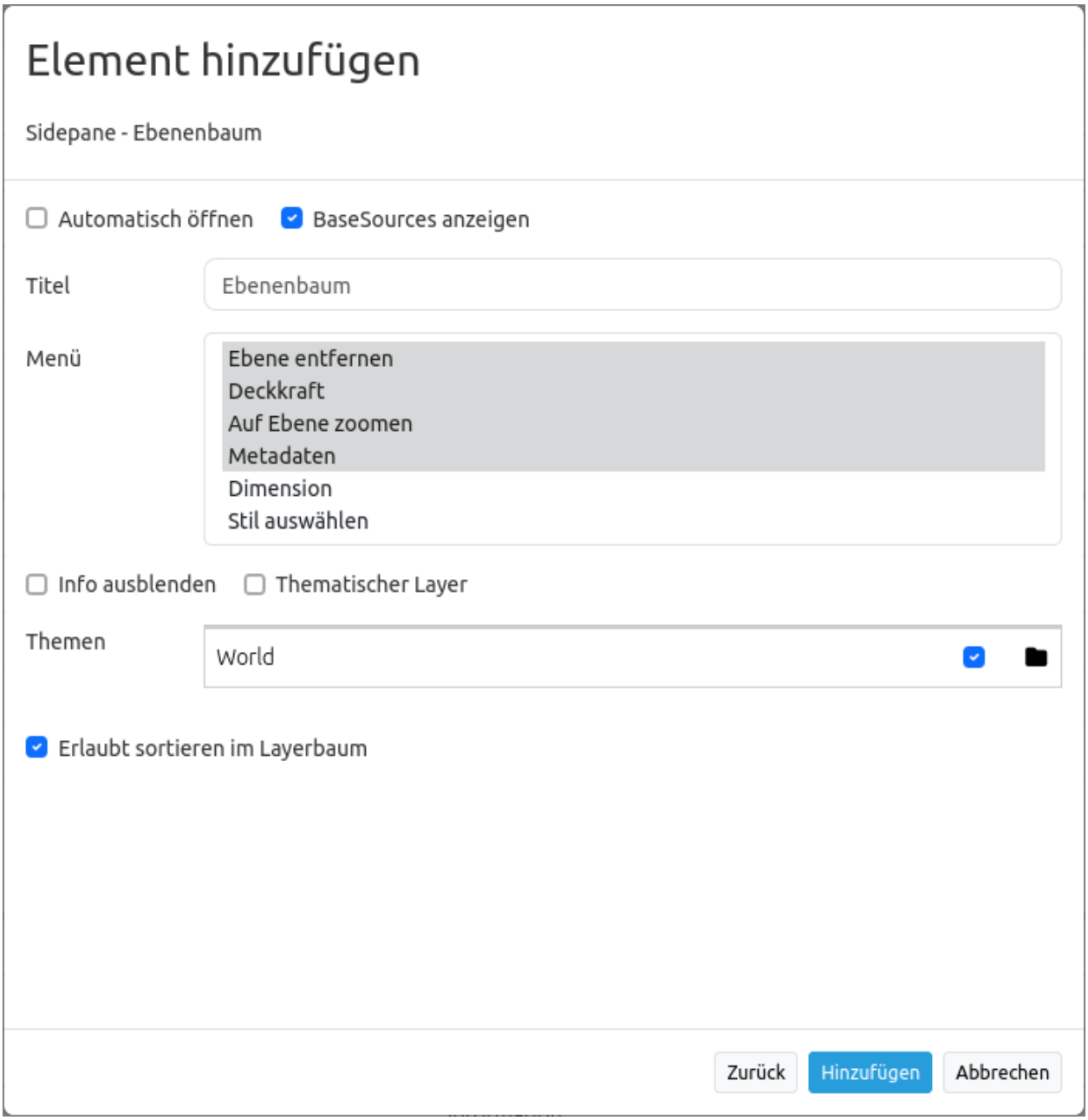


Fig. 7: Einrichtung des Ebenenbaums im Kartenbereich.

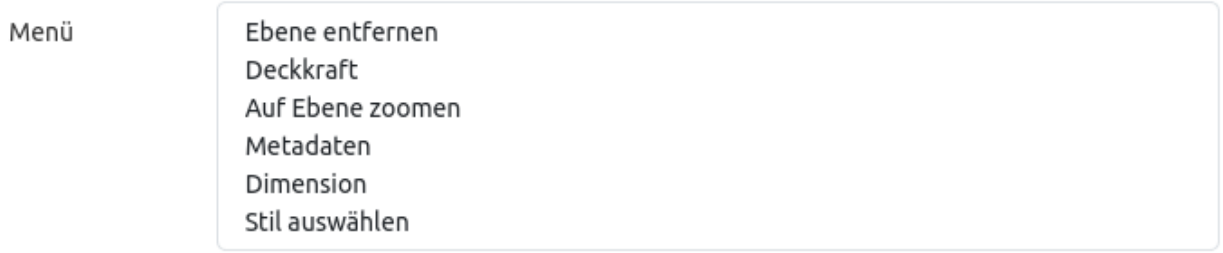


Fig. 8: Konfiguration der Funktionen für das Kontextmenü.

Fig. 9: Kontextmenü der Layer im Ebenenbaum.

Regeln beachtet (wie z.B. maßstabsabhängige Anzeigen).

Hint: Mit der Version 4.1.0 ist es nun möglich, den Status aller Sublayer zu ändern, wenn die Umschalt-Taste (Shift-Taste) während des Umschaltens eines Layers gehalten wird.

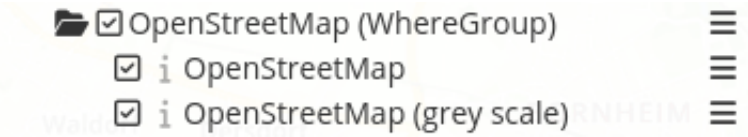


Fig. 10: Ebenenbaum Buttons.

Workflow thematischer Ebenenbaum

Im Folgenden gehen wir eine beispielhafte Konfiguration eines Ebenenbaums mit erweiterten Funktionen in drei Schritten durch:

1. Einrichtung mehrerer Layersets
2. Einrichtung in der Karte zur Anzeige der Layersets
3. Einrichtung des thematischen Ebenenbaums

Einrichtung mehrerer Layersets

In dem folgenden Beispiel sind zwei **Layersets** mit jeweils zwei Instanzen definiert:

- **Layerset Project NRW:**
  - Instanz NW DTK50
  - Instanz Wald und Holz NRW
- **Layerset World:**
  - Instanz OpenStreetMap (OSM)
  - Instanz GEBCO

Für die Einrichtung der Layersets wurden die vier oben genannten Dienste als Instanzen hinzugefügt (detaillierte Information siehe oben oder in der Doku der *Layerset* und *Datenquellen (Sources)*).

Für dieses Beispiel wurden die oben genannten Schritte durchgeführt, um das Layerset “World” mit der Instanz “OpenStreetMap (OSM)” hinzuzufügen. Nun fügen wir in dieses Layerset auch die Instanz “GEBCO” hinzu. Um die thematische Gruppierung nutzen zu können, erstellen wir zusätzlich ein neues Layerset mit dem Namen “Project NRW” und laden in dieses die beiden oben genannten Instanzen “NW DTK50” und “Wald und Holz NRW” ein.

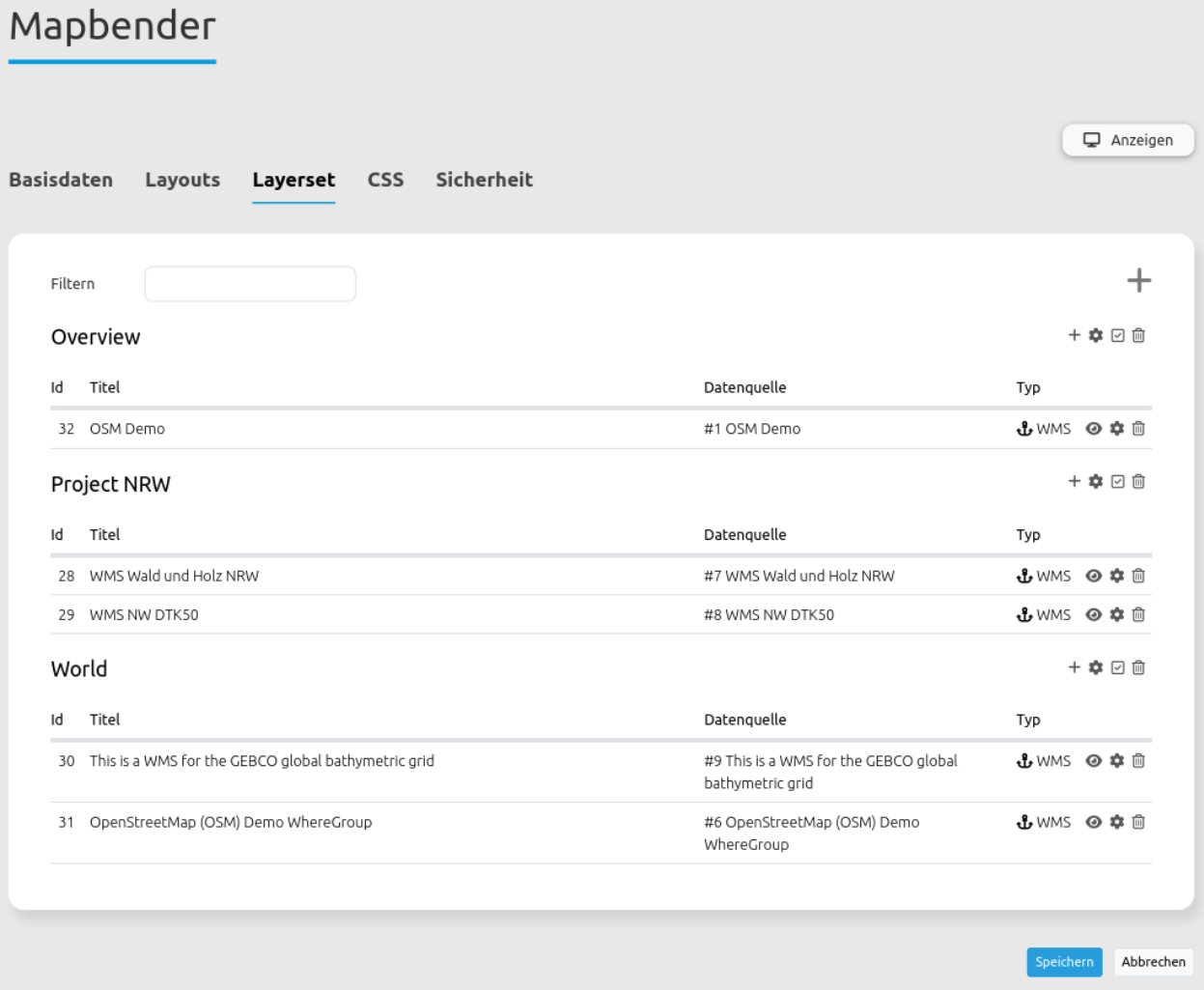


Fig. 11: Konfiguration der Layersets für den thematischen Ebenenbaum.

Das Layerset sollte nun drei Layersets enthalten. Die **Overview** für die Übersichtskarte, das **Project NRW** Layerset mit den zwei regionalen Datensets aus NRW und das "World"-Layerset mit den überregionalen Layern.

Einrichtung in der Karte zur Anzeige der Layersets

Als Nächstes erfolgt die Einrichtung des *Kartenelements* zur Anzeige des Layersets in der Karte. Dazu wechseln wir in den Reiter *Layouts* und bearbeiten das Kartenelement im Kartenbereich. Wichtig ist, dass bei dem Bereich Layersets beide Layersets "World" und "Project NRW" per Checkbox aktiviert sind, damit diese in der Anwendung angezeigt werden.

Bei Fragen zur weiteren Konfiguration der Karte kann die Dokumentation des *Kartenelements* weiterhelfen.

Einrichtung des thematischen Ebenenbaums

Als letzter Schritt erfolgt die Einrichtung des Ebenenbaums selbst. Für den thematischen Ebenenbaum binden wir den Ebenenbaum in diesem Beispiel in die Sidepane ein.

Ist die Option **Thematischer Layer** ausgeschaltet, benutzt der Ebenenbaum nicht die konfigurierten Layersets und zeigt die einzelnen Instanzen ohne thematische Strukturierung in der Hauptebene an. Nun wollen wir jedoch die Layer über unsere thematischen Layersets anzeigen, daher aktivieren wir die Funktion **Thematischer Layer**. Da wir in der *Karte (Map)* beide Layersets in die Anwendung eingebunden haben, werden diese nun unter dem **Themen**-Bereich angezeigt.

Damit die **Themen** in der Anwendung wie gewünscht angezeigt werden, gibt es verschiedene Einstellungsmöglichkeiten:

[1] Thema anzeigen Ist diese Option gesetzt, wird das Layerset als zusätzliche Ebene angezeigt. Ist diese Option nicht gesetzt, werden die enthaltenen Layer-Instanzen in der Hauptebene angezeigt.

[2] Thema offen oder geschlossen Ist diese Option gesetzt (Symbol des geöffneten Ordners), ist das Thema im Ebenenbaum automatisch ausgeklappt (sofern "Thema anzeigen" auch aktiv ist).

Wenn wir im Thema "World" die Standardeinstellungen beibehalten und im Thema "Project NRW" die anderen Optionen aktivieren, sieht die Konfiguration des Elements wie folgt aus:

Wir haben die Layersets somit als thematische Gruppen in den Ebenenbaum eingebunden. Durch die Konfiguration der thematischen Layer stellt sich der Ebenenbaum in der Anwendung nun wie folgt dar:

Das Layerset "World" wird als Thema angezeigt, ist jedoch nicht geöffnet. Beim Layerset "Project NRW" wird das Thema beim Öffnen der Anwendung aufgeklappt gezeigt. Zusätzlich können alle Layer können über einen Button aktiviert werden.

YAML-Definition

Diese Vorlage kann genutzt werden, um das Element in einer YAML-Anwendung einzubinden.

```
title: layertree                                    # Titel des Ebenenbaums
target: ~                                           # ID des Kartenelements
type: ~                                             # Typ des Ebenenbaums (Element oder
↪Dialog)
autoOpen: false                                     # Öffnet den Ebenenbaum beim
↪Anwendungsstart (Standard: false)
showBaseSource: true                                # Zeigt den Basislayer an (Standard:
↪true)
showHeader: true                                    # Zeigt eine Überschrift, die die Anzahl
↪der Services zählt (Standard: true)
```

(continues on next page)

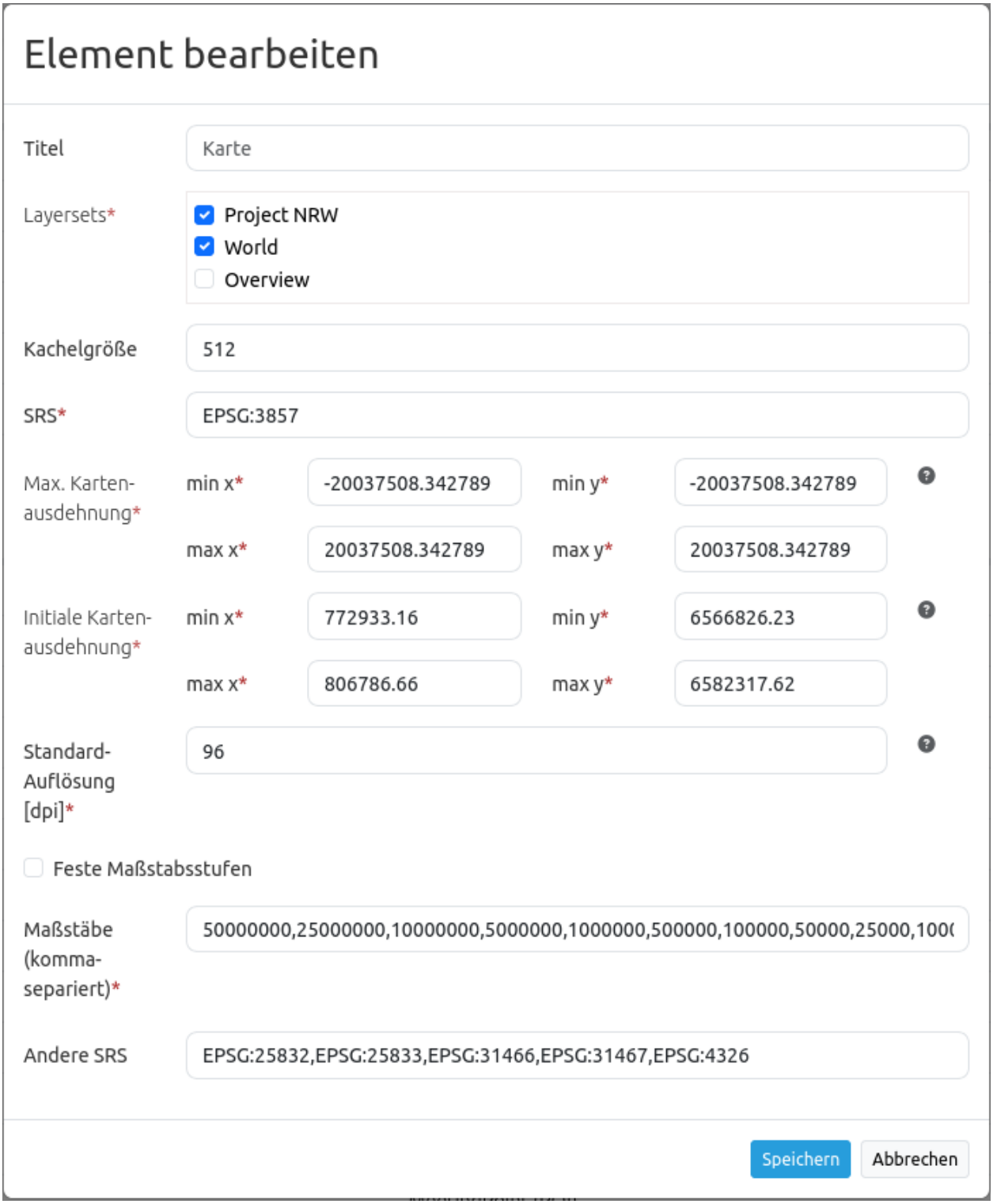


Fig. 12: Konfiguration des Kartenelements für den thematischen Ebenenbaum.

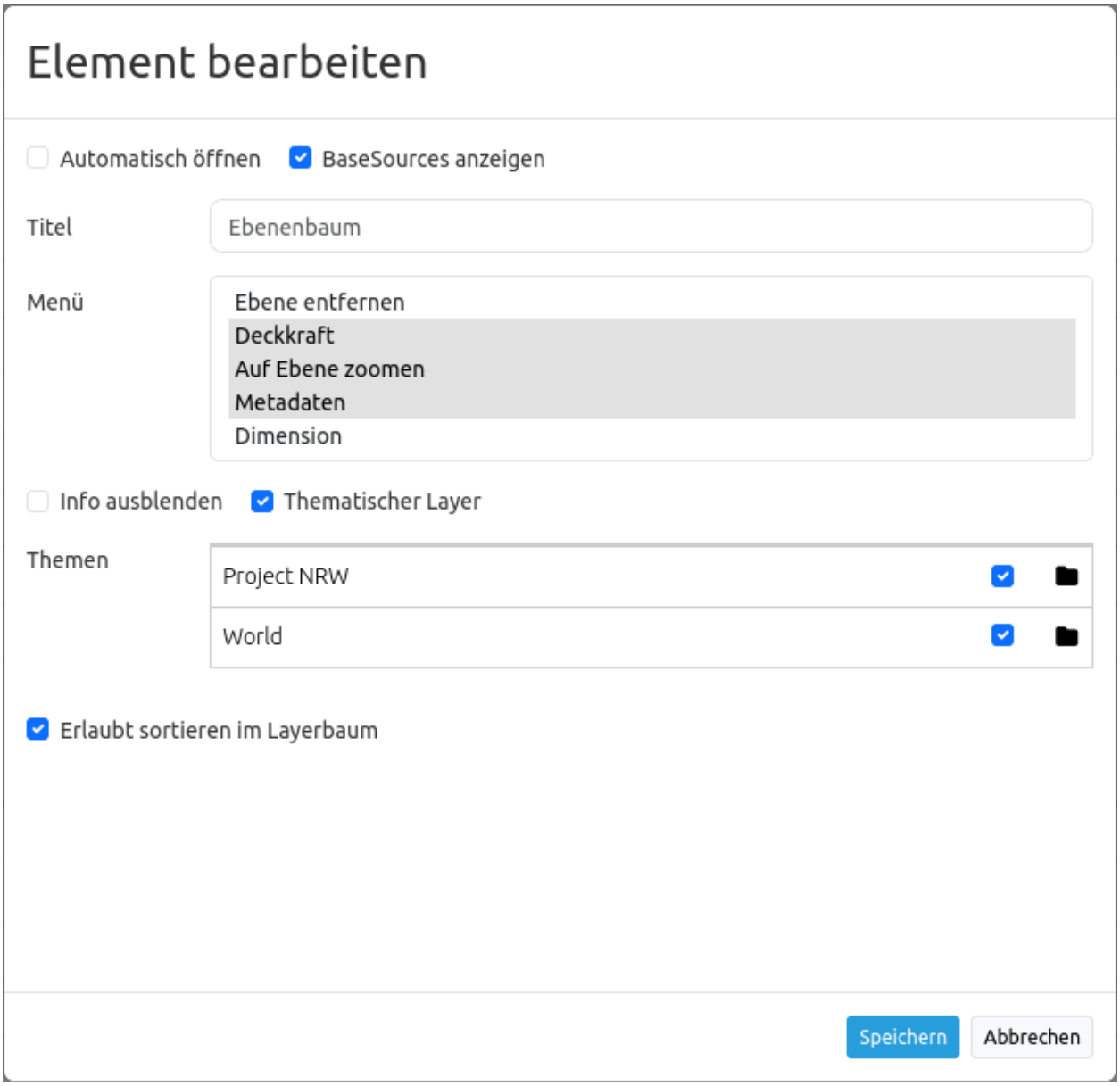


Fig. 13: Einrichtung des thematischen Ebenenbaums im Kartenbereich.

Fig. 14: Backend-Konfiguration Thematische Layer.

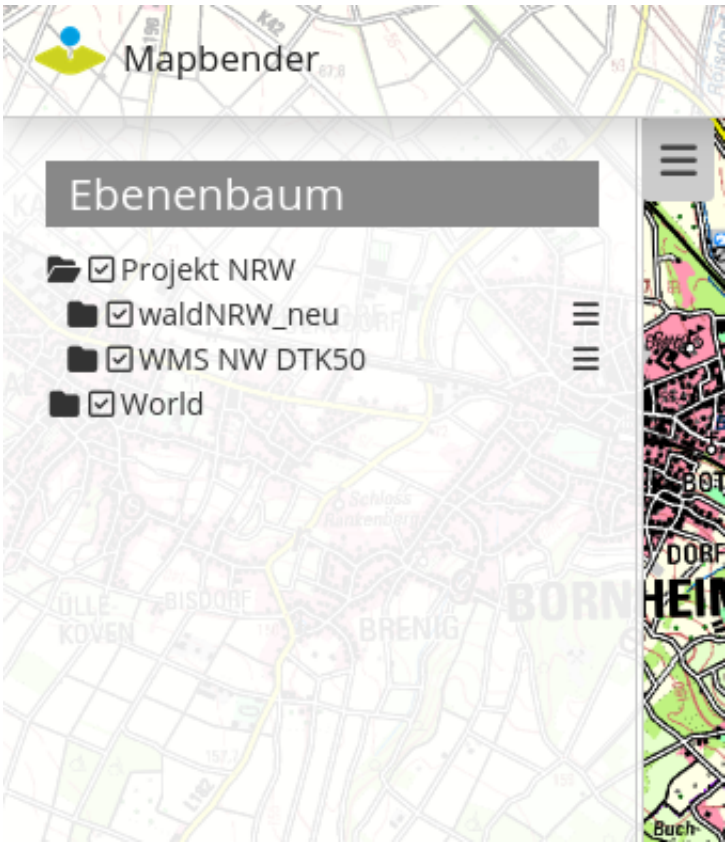


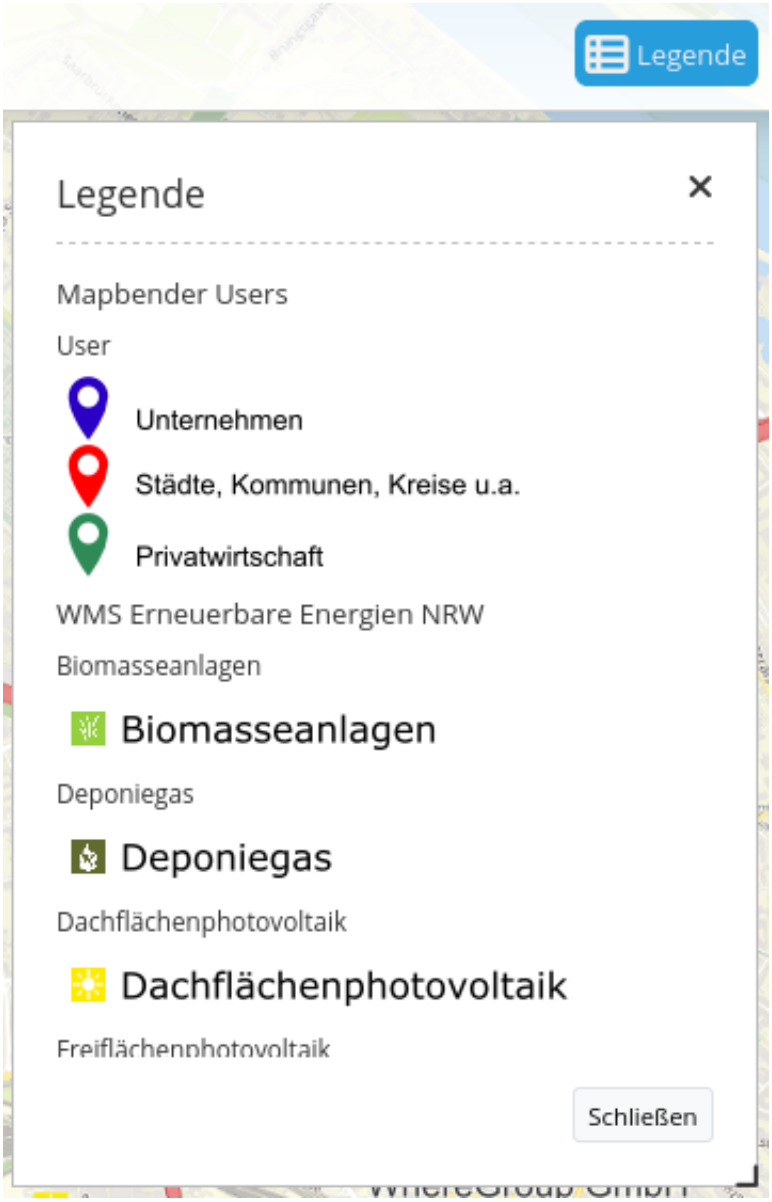
Fig. 15: Aufbau des thematischen Layersets in der Sidepane.

(continued from previous page)

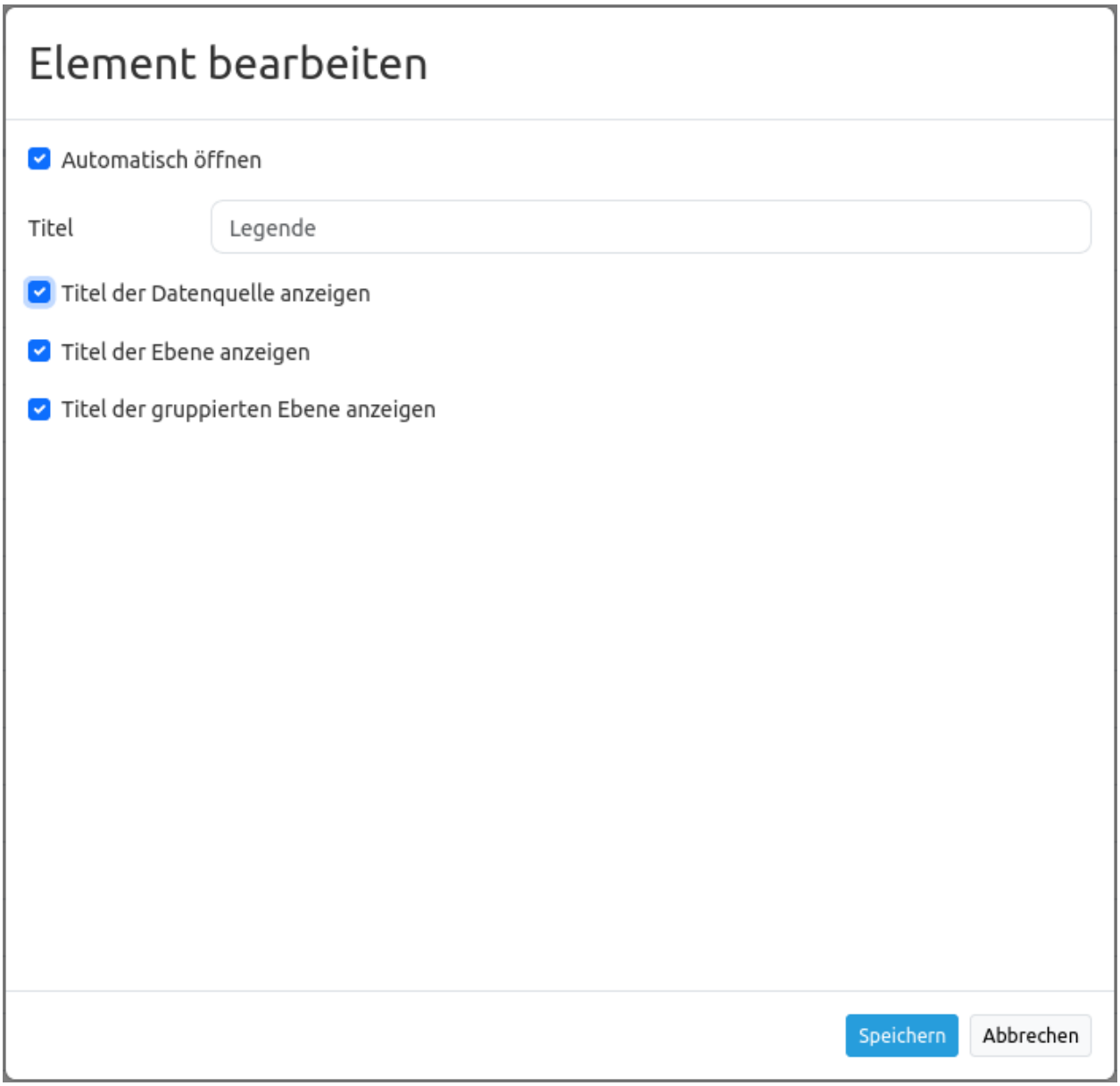
```
menu: [opacity,zoomtolayer,metadata,removelayer]    # Zeigt ein Kontextmenü für den Layer an
↪(wie Transparenz, Zoom auf Layer, Anzeige des Metadatendialogs, Layer entfernen)
hideInfo: null
hideSelect: null
allowReorder
themes: {  }
```

3.1.7 Legende (Legend)

Dieses Element zeigt eine Legende der Layer an, die in der Karte dargestellt werden. Nicht jeder Layer verfügt dabei über eine Legende. Zusätzlich zum Legendenbild können der Titel des Dienstes, des Layers und der Gruppe (sofern vorhanden) angezeigt werden.



Konfiguration



- **Automatisches Öffnen:** Sofern aktiviert, öffnet sich die Legende beim Anwendungsstart (Standard: deaktiviert).
- **Titel:** Titel des Elements. Dieser wird in der Layouts Liste angezeigt. In der Karte wird der Titel neben dem *Button* angezeigt, wenn dort "Beschriftung anzeigen" aktiviert ist.
- **Titel der Datenquelle anzeigen:** zeigt den WMS Titel (Standard: true).
- **Titel der Ebene anzeigen:** zeigt den Layertitel (Standard: true).
- **Titel der gruppierten Ebenen anzeigen:** zeigt den Gruppenlayertitel für gruppierte Layer (Standard: true).

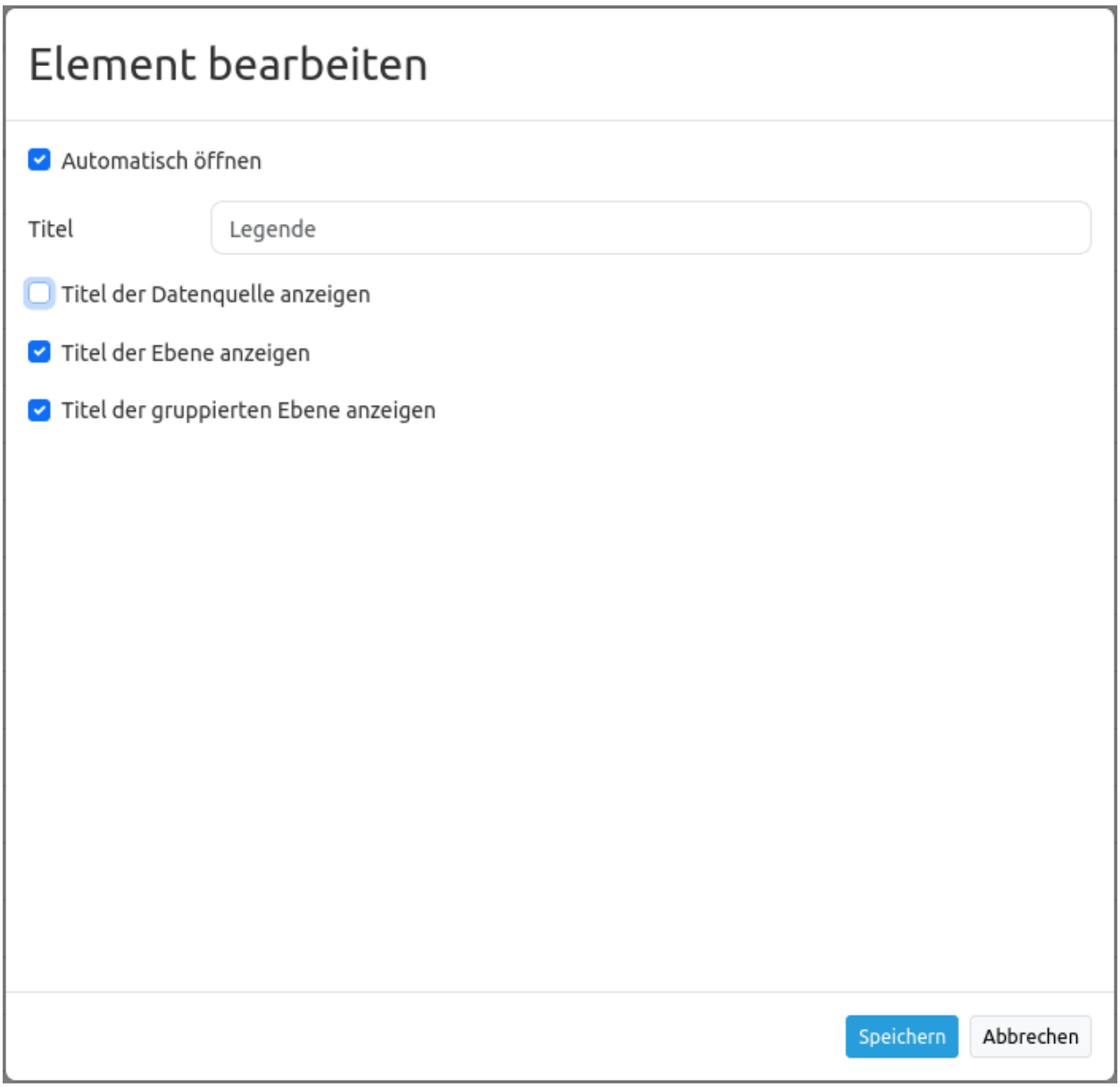
Für das Element wird ein Button oder die Sidepane verwendet. Zu der Konfiguration des Buttons besuchen sie die Dokumentationsseite unter *Button*.

Konfigurationsbeispiele

Legende in der Sidepane

Die Legende in der Sidepane wird über + im *Backend* der Anwendung unter Layouts - Sidepane eingebunden.

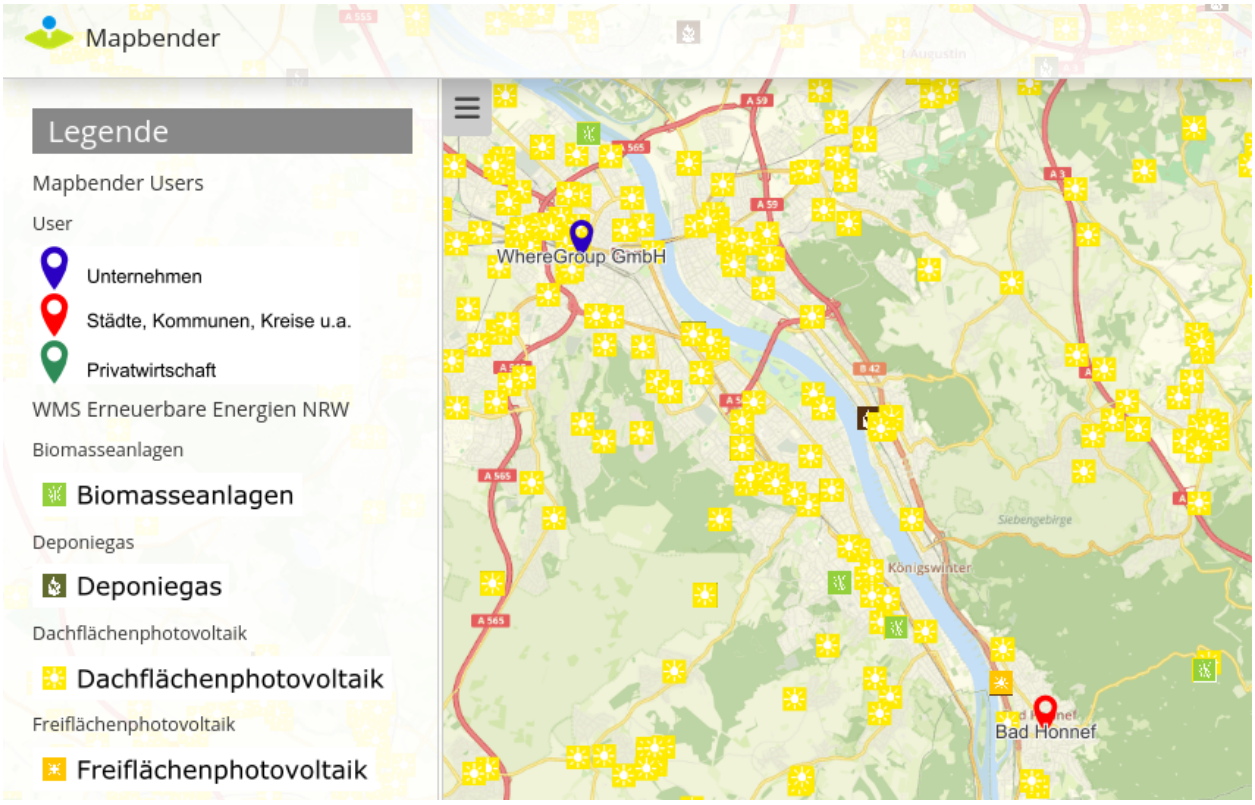
Anschließend wird das Element ausgewählt. Es öffnet sich der Konfigurationsdialog.



Das hier konfigurierte Element hat den Titel “Legende”. Die Legende öffnet sich automatisch (Häkchen bei *Automatisches Öffnen*). Der Titel der Ebenen sowie der Titel der gruppierten Ebenen (Häkchen bei

Titel der Ebene anzeigen und *Titel der gruppierten Ebene anzeigen*) werden angezeigt.

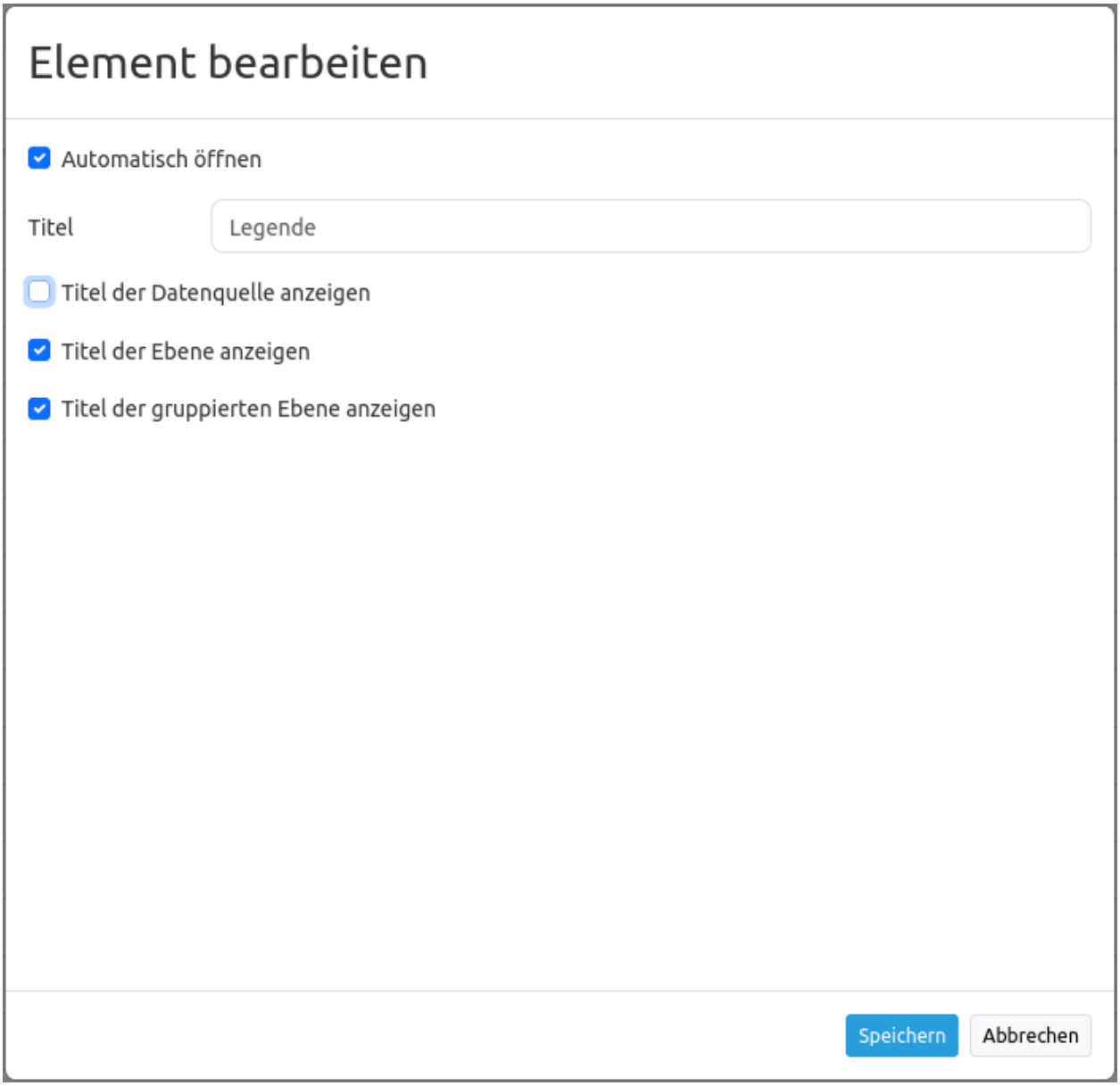
Diese Konfiguration ergibt folgendes Ergebnis in der Anwendung:



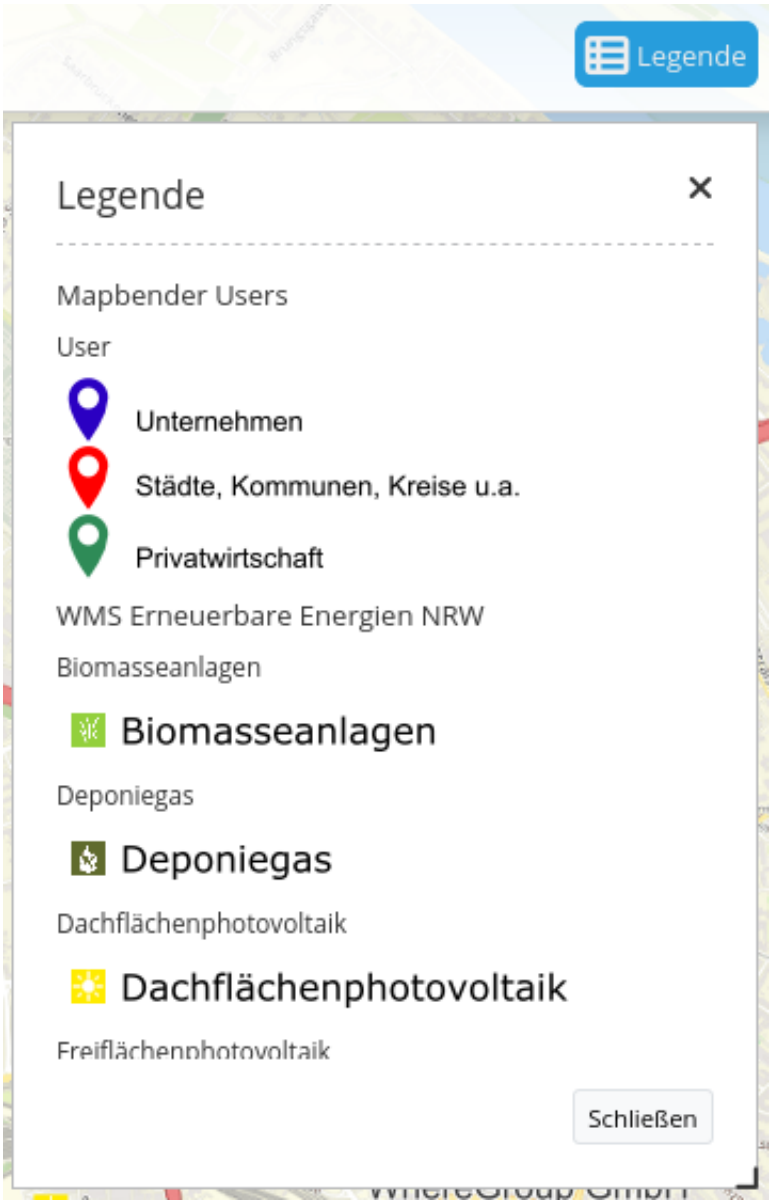
Legende in der Oberen Werkzeugleiste (Toolbar)

Die Legende kann auch als Button in der Werkzeugleiste eingebunden werden. Hierfür muss zuerst das Element Legende unter dem Reiter Layouts im Kartenbereich eingefügt werden.

Für das Konfigurationsbeispiel wurden folgende Einstellungen gewählt:

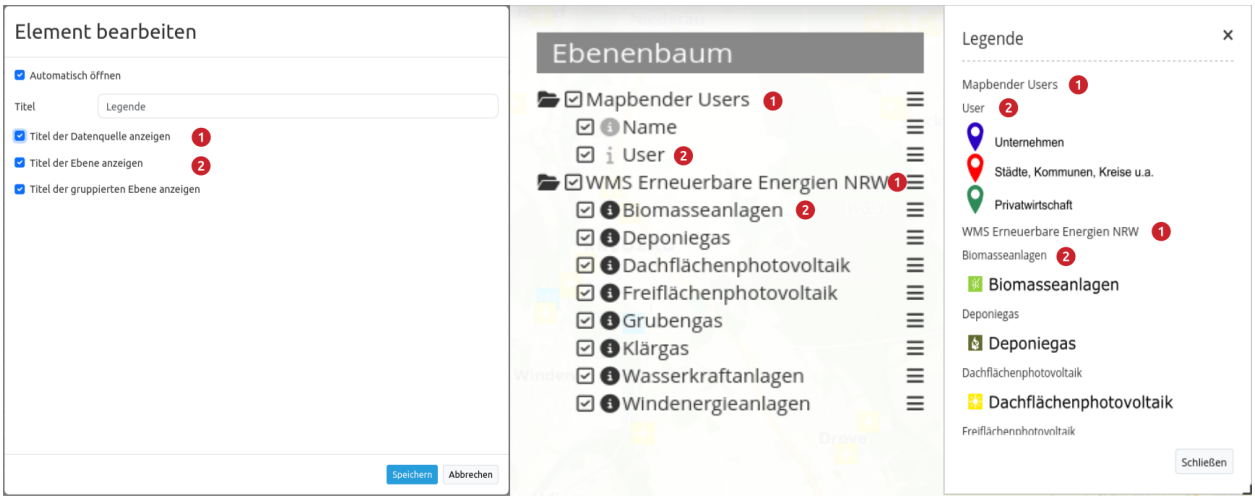


Sobald dieses Element im Kartenbereich eingebunden wurde, muss ein Button in der oberen Werkzeugleiste eingefügt werden. Die Konfiguration von Buttons wird in der Mapbender-Dokumentation unter *Button* beschrieben. Mit diesen Einstellungen sieht das Ergebnis in der Anwendung wie folgt aus:



In der Oberen Werkzeugleiste (Toolbar) ist der Button für das Legenden-Element zu sehen. Sobald auf den Button geklickt wird, öffnet sich ein Dialog mit der Legende. Durch die Einstellung “Automatisch öffnen” wird der Dialog beim Start der Anwendung direkt angezeigt.

Inwiefern sich die Aktivierung bzw. Deaktivierung einzelner Haken auf die Legende auswirkt, ist hier zu sehen:



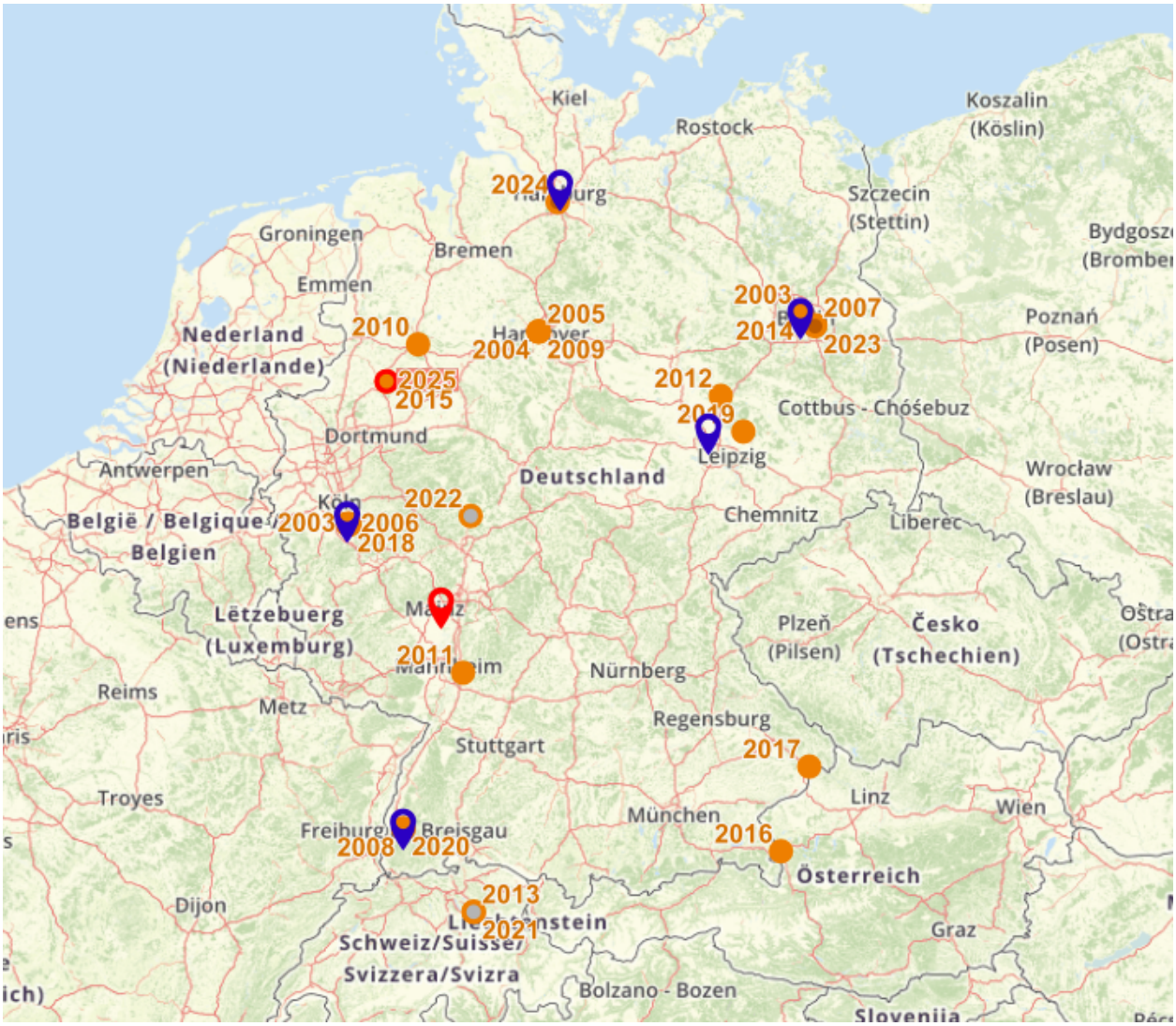
YAML-Definition

Diese Vorlage kann genutzt werden, um das Element in einer YAML-Anwendung einzubinden.

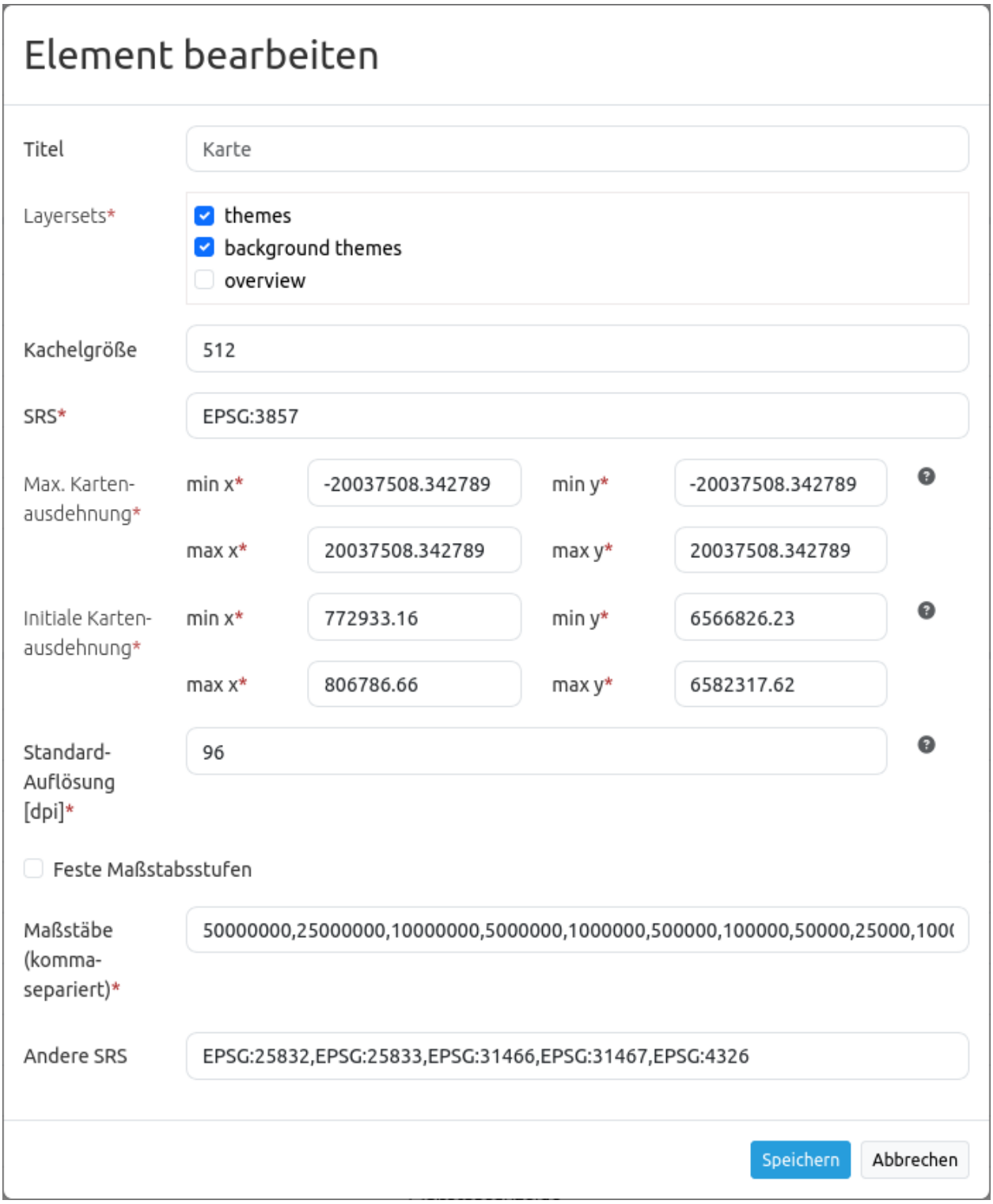
```
tooltip: 'Legend'                    # Text des Tooltips
elementType: dialog                  # Anzeige als Dialog- oder Blockelement (Standard: Dialog)
autoOpen: true                       # true, wenn die Legende beim Start der Anwendung geöffnet werden
↪soll (Standard: false)
displayType: list                    # Akkordeonartige Anzeige oder Liste (Standard: Liste)
target: ~                            # ID des Kartenelements
hideEmptyLayer: true                 # true/false Layer wird versteckt, wenn keine Legende verfügbar
↪ist (Standard: true)
showWmsTitle: true                   # true/false zeigt den WMS Titel (Standard: true)
showLayerTitle: true                 # true/false zeigt den Layertitel (Standard: true)
showGroupedLayerTitle: true          # true/false zeigt den Gruppentitel für gruppierte Layer
↪(Standard: true)
```

3.1.8 Karte (Map)

Die Karte ist das zentrale Element einer Anwendung. Sie basiert auf OpenLayers und wird als Element in den Kartenbereich im *Backend* integriert.


Kiel
Rostock
Koszalin
(Köslin)
2024
Szczecin
(Stettin)
Groningen
Bremen
Bydgosz
(Brombe
Emmen
2003
2007
2010
2005
Nederland
(Niederlande)
2004
2009
2014
2023
Poznań
(Posen)
2025
2015
2012
2019
Cottbus - Chóśebuz
Dortmund
Leipzig
Antwerpen
Deutschland
Wrocław
(Breslau)
2022
Chemnitz
België / Belgique
2003
2006
Liberec
2018
Belgien
Lëtzebuerg
(Luxemburg)
Plzeň
(Pilsen)
Česko
(Tschechien)
2011
Nürnberg
Reims
Metz
Regensburg
Stuttgart
2017
Troyes
Linz
Wien
München
Freiburg
Breisgau
2008
2020
2016
Österreich
2013
Dijon
2021
Graz
Schweiz/Suisse/
Svizzera/Svizra
Bolzano - Bozen

Konfiguration



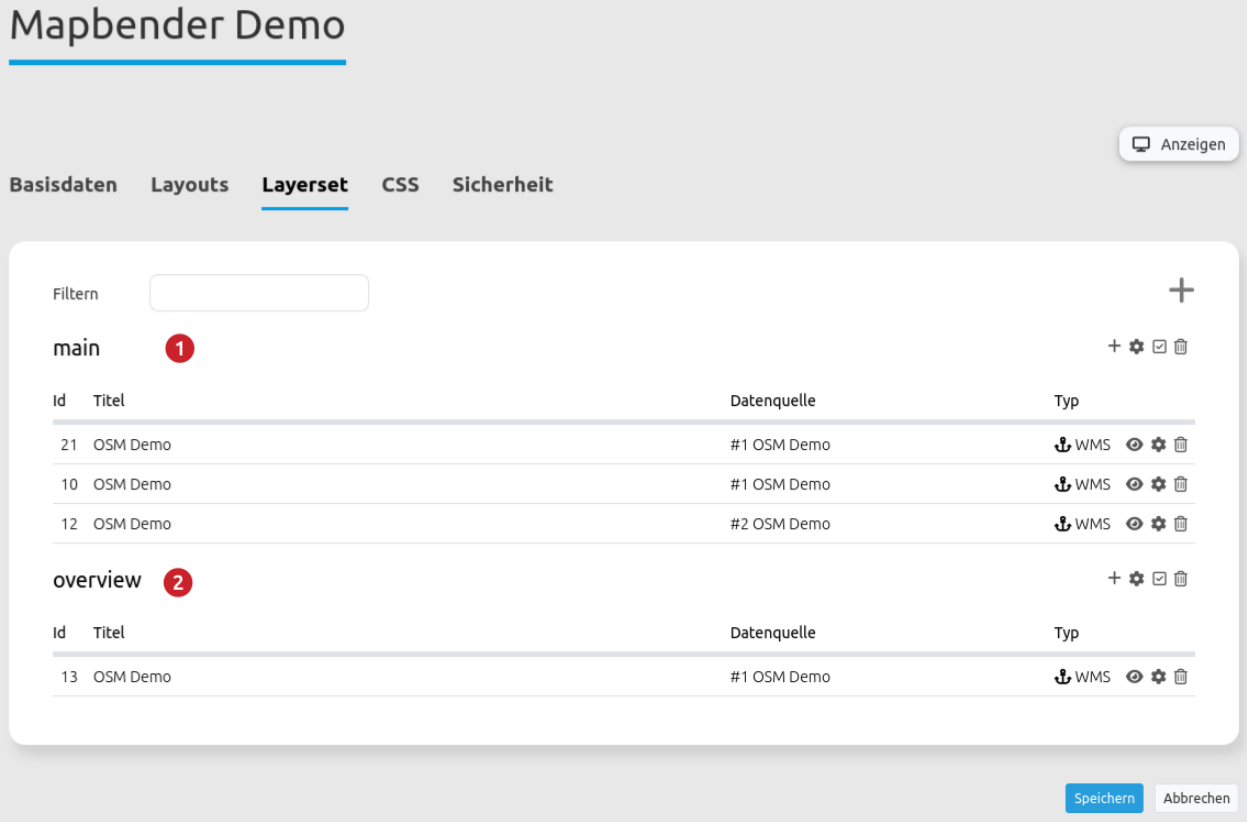
- **Titel:** Titel des Elements. Dieser wird im *Layouts*-Bereich angezeigt.
- **Layersets:** Wählen Sie ein oder mehrere *Layerset* aus, die vom Kartenelement dargestellt werden sollen. Dabei kann die Reihenfolge der Layersets via Drag-and-drop verändert werden.

- **Kachelgröße:** Größe der Tiles bei gekachelten Diensten.
- **SRS:** Koordinatenbezugssystem beim Start der Anwendung ("Spatial Reference System"). Zwei Arten der SRS Definition werden unterstützt: EPSG:CODE oder EPSG:CODE|MEIN SRS TITEL. Wird kein eigener SRS-Titel definiert, so wird die Standarddefinition aus der Title-Spalte der Tabelle *mb_core_srs* verwendet.
- **Max. Kartenausdehnung:** Maximaler Kartenbereich (BBOX mit min/max x/y). Ein Umgebungsrechteck, das den Ausschnitt definiert. Bei Klick auf das Globus-Icon im *Navigationswerkzeug (Navigation Toolbar)* wird auf die maximale Ausdehnung gezoomt.
- **Initiale Kartenausdehnung:** Bereich der Karte, der beim Starten der Anwendung angezeigt wird (BBOX mit min/max x/y). Ein Umgebungsrechteck, das den Ausschnitt definiert. Bei Klick auf das Haus-Icon im *Navigationswerkzeug (Navigation Toolbar)* wird auf die initiale Ausdehnung gezoomt.
- **Standard-Auflösung [dpi]:** Die Auflösung passt sich auf Basis dieses Wertes an die Auflösung des verwendeten Gerätes an. Standard: 96 dpi.
- **Feste Maßstabsstufen:** Das Zoom-Verhalten wird hierdurch konfiguriert. Feste Maßstabsstufen verbessern die visuelle Qualität von Diensten, welche nur auf bestimmten Maßstäben zwischengespeichert werden. Ist die Einstellung aktiviert, dann können nur Maßstäbe ausgewählt werden, die auch unter *scales* im Folgenden definiert sind (Standard: false).
- **Maßstäbe (csv):** Festgelegte Zoomstufen, die durch Drehen des Mausrads oder bei stufenweisem Zoomen für den Maßstab genutzt werden (werden durch Komma getrennt).
- **Andere SRS:** Weitere auswählbare Projektionen unter denen die Karte angezeigt werden kann (werden durch Komma getrennt). Zwei Arten der SRS Definition werden unterstützt: EPSG:CODE oder EPSG:CODE|MEIN SRS TITEL.

Konfigurationsbeispiel

Das Kartenelement (Map) muss unter Layouts im Kartenbereich integriert werden:

In der Karte können alle Instanzen angezeigt werden, welche im Layerset enthalten sind. Im vorliegenden Beispiel wird zwischen der *Hauptkarte/main* (1) und *Übersichtskarte/overview* (2) unterschieden.



Damit *Layersets* auch in der Karte angezeigt werden, müssen diese im Kartenelement aktiviert werden. Eine Mehrfachauswahl ist hierbei auch möglich. Layersets, welche nicht ausgewählt wurden, können als *Übersichtskarte (Overview)* fungieren. Im Beispiel dient *main* als Haupt- und *overview* als Übersichtskarte.

Im Feld *SRS* muss das Koordinatenreferenzsystem eingetragen werden. Im Beispiel ist dies EPSG:3857 bzw. WGS 84/Pseudo-Mercator. Wenn andere Referenzsysteme zur Auswahl stehen sollen, werden diese unter *Other SRS* angegeben. Im Beispiel umfasst dies: EPSG:25832 (ETRS89/UTM Zone 32N), EPSG:25833 (ETRS89/UTM Zone 33N), EPSG:31466 (DHDN/3-degree Gauss-Krüger Zone 2), EPSG:31467 (DHDN/3-degree Gauss-Krüger Zone 3) und EPSG:4326 (WGS 84).

Weiterhin muss das Feld *Max. Kartenausdehnung* definiert werden. Dieses gibt den maximalen sichtbaren Kartenbereich an. Alles was außerhalb dieser Angabe liegt, wird in der Anwendung nicht dargestellt. Das Pflichtfeld *Initiale Kartenausdehnung* wiederum definiert den Startbereich, welcher beim Öffnen der Anwendung zu sehen ist. Im Beispiel wird das Stadtgebiet von Bonn angezeigt.

Die Standard-Auflösung in dpi definiert die Auflösung des verwendeten Geräts; der zugehörige Standardwert von 96 dpi ist über dieses Feld anpassbar. Falls die dargestellte Auflösung der Karte nicht mit der des WMS-Dienstes übereinstimmt, kann eine Veränderung des Werts helfen, um die Karte passend darzustellen.

Note: Hinweis: Die maßstabsabhängige Anzeige funktioniert derzeit nur auf Desktops mit regulärer Auflösung zuverlässig.

Zuletzt definiert *Scales (csv)* die unterschiedlichen Maßstabsstufen in der Anwendung. Zwischen diesen kann mithilfe des *Maßstabsauswahl (Scale selector)* oder der *Navigationswerkzeug (Navigation Toolbar)* navigiert werden. Feste Maßstabsstufen wurden dabei im Beispiel deaktiviert, weshalb auch eine Auswahl anderer Maßstäbe über das Mausrad möglich ist.

YAML-Definition

Diese Vorlage kann genutzt werden, um die Karte in einer YAML-Anwendung einzubinden.

```
layerset: null              # Definiert und verweist auf ein Layerset
srs: "EPSG:4326"            # Koordinatenbezugssystem. Zwei Arten der SRS Definition werden unterstützt:
                              - "EPSG:CODE" oder
                              - "EPSG:CODE|MEIN SRS TITEL"
extents:
    max: [0, 40, 20, 60]
    start: [5, 45, 15, 55] # Kartenbereich (extent)
scales: "25000000,10000000,5000000,1000000,500000" # eine CSV-Liste für den Maßstab
otherSrs:
    - EPSG:31466
    - EPSG:31467
    - EPSG:25832           # andere Koordinatenbezugssystem. Zwei Arten der SRS Definition werden
↪unterstützt:
                              - ["EPSG:CODE","EPSG:CODE"] or
                              - ["EPSG:CODE|MEIN SRS TITEL","EPSG:CODE|MEIN SRS TITEL"]
tileSize: 256               # Kachelgröße
```

Kontrolle über URL-Parameter

Ebenen aktivieren

Mapbender ermöglicht über den URL-Parameter `visiblelayers` die Möglichkeit, Layer unabhängig von ihrer Backend-Konfiguration beim Start einer Anwendung zu aktivieren. Hierbei kann die Aktivierung entweder via *ID* oder *Name* erfolgen:

- **ID**: <InstanceID>/<InstanceLayerID>
- **Name**: <RootLayerName>/<LayerName>

InstanceID/InstanceLayerID: Auf diese Weise werden die anwendungsspezifischen Werte von InstanceID und InstanceLayerID übergeben:

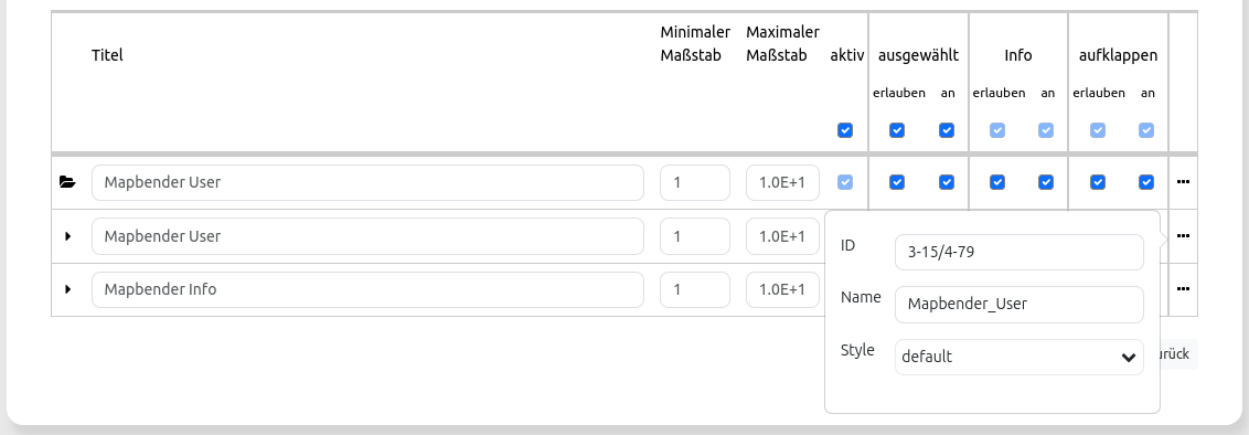
```
?visiblelayers=<InstanceID>/<InstanceLayerID>
```

RootLayerName/LayerName: Auf diese Weise werden Ebenen entlang der Kombination aus RootLayer- und Layernamen als Parameter übergeben:

```
?visiblelayers=<RootLayerName>/<LayerName>
```

Hint: Bitte beachten Sie, dass sich die IDs nach jedem Dienst-Update ändern. Die Übergabe des Names ist daher ggf. die konstantere Lösung.

Um weitere Layereigenschaften anzuzeigen, gibt es im Layerset-Reiter bei der Instanz-Konfiguration neben jedem Layer ein Icon mit drei Punkten. Klicken Sie auf das Icon, damit ein Informationsfenster erscheint:



- **ID**: Der erste Wert im oberen Textfeld nennt die interne *SourceID* und die *SourceLayerID* (3-15). Der zweite Wert im oberen Textfeld nennt die *InstanceID* und die *InstanceLayerID* (4-79).
- **Name**: Im zweiten Textfeld steht der *LayerName*. Dabei wird in der ersten Zeile der *RootLayerName* ausgegeben.
- **Style**: Im dritten Dropdownfeld können Stylingalternativen ausgewählt werden, sofern verfügbar.

Nutzen Sie für eine ID-Übergabe die *zweite* Wertkombination nach dem Schrägstrich für den `visiblelayers`-Parameter in der URL. Trennen Sie die beiden Werte mit einem Schrägstrich (anstelle des Bindestrichs):

Zum Beispiel: `https://localhost/mapbender/application/myapp?visiblelayers=4/79`

Zwei oder mehr nicht-sichtbare Layer können kommasepariert übergeben werden. Fügen Sie dazu die jeweiligen Attributwerte nach demselben Schema ein:

Zum Beispiel: `https://localhost/mapbender/application/myapp?visiblelayers=4/79,1/42`

Auch Kombinationen aus Namen und ID-Werten sind möglich:

`https://localhost/mapbender/application/myapp?visiblelayers=Mapbender/Mapbender_Names,Mapbender/Mapbender_User,39/149`

POI übergeben

Sie können beim Aufruf mit der URL Koordinaten übergeben. Die damit verknüpfte Position wird anschießend auf der Karte mithilfe eines (optional beschriftbaren) POI markiert.

Jeder POI verfügt dabei über die folgenden Parameter:

- Punkt (point): Koordinate, die Werte werden mit Komma getrennt (Pflichtfeld),
- Beschriftung (label): Beschriftung, die angezeigt werden soll (optional),
- Maßstab (scale): Maßstab, in dem der Punkt angezeigt werden soll (optional).

Format für die Übergabe eines Punktes:

```
?poi[point]=368777,5619411&poi[label]=Rheinaue&poi[scale]=10000
```

Rechteck (BBOX) übergeben

Es kann ein Rechteck (BBOX) beim Start übergeben werden. Es wird dann auf dieses Rechteck gezoomt. Der Aufruf sieht wie folgt aus:

```
?bbox=364286,5622263,365979,5622806
```

Maßstab übergeben

Es kann der gewünschte Maßstab aus der Liste der verfügbaren Maßstäbe übergeben werden.

```
?scale=1000
```

SRS (Projektion) übergeben

Es kann eine gewünschte Projektion für den Start der Anwendung übergeben werden.

```
?srs=EPSG:4326
```

Anwendung über Koordinate zentrieren

Es kann eine Koordinate beim Start übergeben werden, die in der Anwendung zentriert werden soll.

```
?center=364286,5622263
```

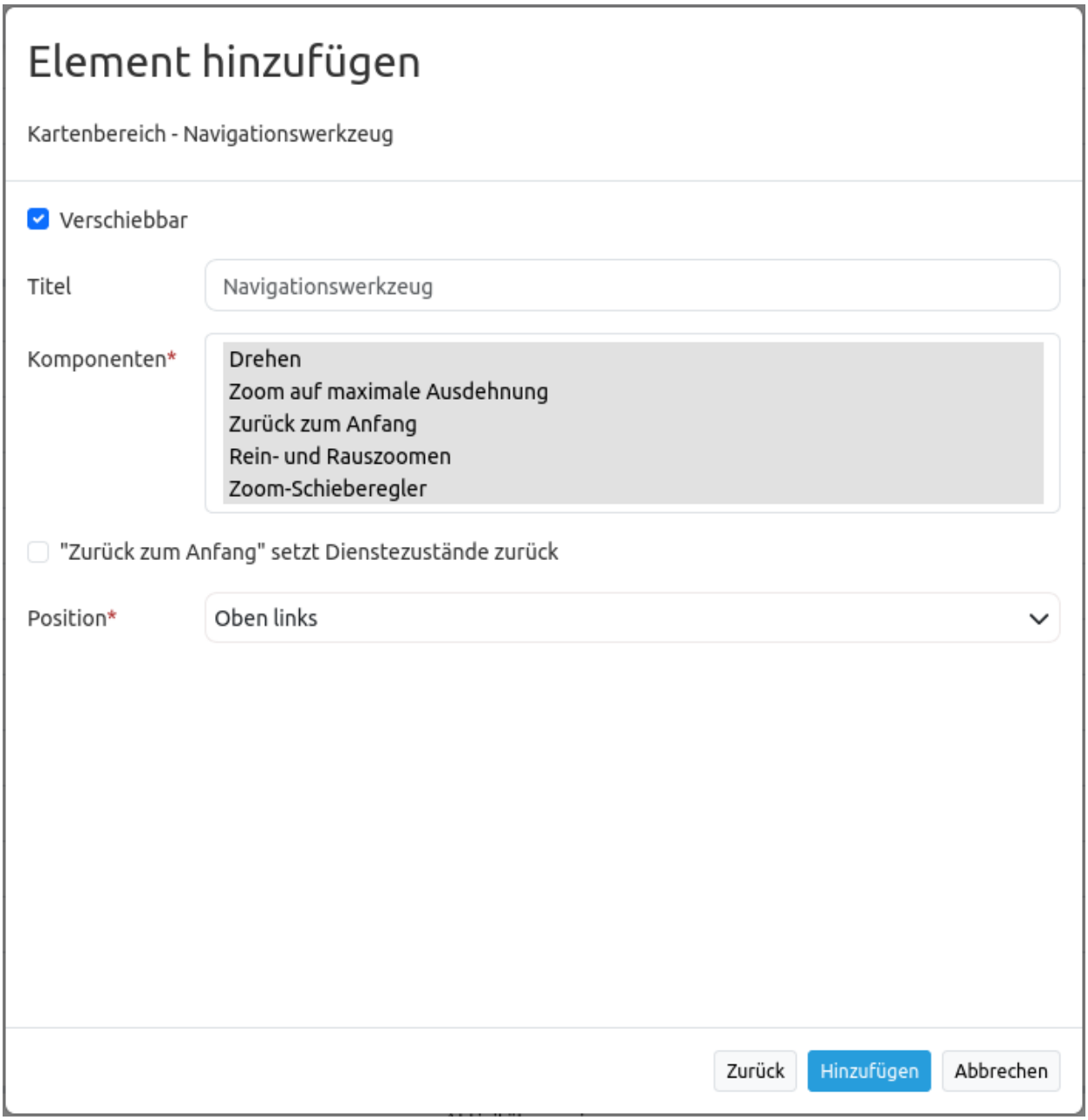
Weitere Startparameter

Die Elemente WMS Loader und WMC Loader stellen ebenfalls Parameter zur Verfügung, die beim Start einer Anwendung übergeben werden können. Schauen Sie sich für nähere Informationen die Dokumentation dieser Elemente an.

3.1.9 Navigationswerkzeug (Navigation Toolbar)

Das Element erleichtert die Navigation in der Karte (Rotation und Zoom) durch entsprechende Buttons. Der Kartenmaßstab kann dabei individuell über eine Leiste angesprungen oder über die Zoom in/out-Buttons abgeändert werden. Es besteht zudem die Möglichkeit, auf die maximale Kartenausdehnung zu zoomen oder zum Anfangszustand zurück zu navigieren. Das Navigationselement kann verschoben werden, wenn die Option aktiviert wird.

Konfiguration



- **Verschiebbar:** Navigationswerkzeug ist verschiebbar oder nicht (Standard: true).
- **Titel:** Titel des Elements. Dieser wird in der Layouts Liste angezeigt.
- **Komponenten:** Komponenten des Navigationswerkzeugs; Pflichtfeld. Auswahlmöglichkeiten: Drehen, Zoom auf maximale Ausdehnung, Zurück zum Anfang, Rein- und Rauszoomen und Zoom-Schieberegler.
- **“Zurück zum Anfang” setzt Dienstezustände zurück:** Zurücksetzen von Dienstzuständen (Standard: false).
- **Position:** Ausrichtung des Navigationswerkzeugs (Standard: ‘left-top’); Pflichtfeld. Auswahlmöglichkeiten: inline (zum Einbinden in der Sidepane), left-top (oben links), left-bottom

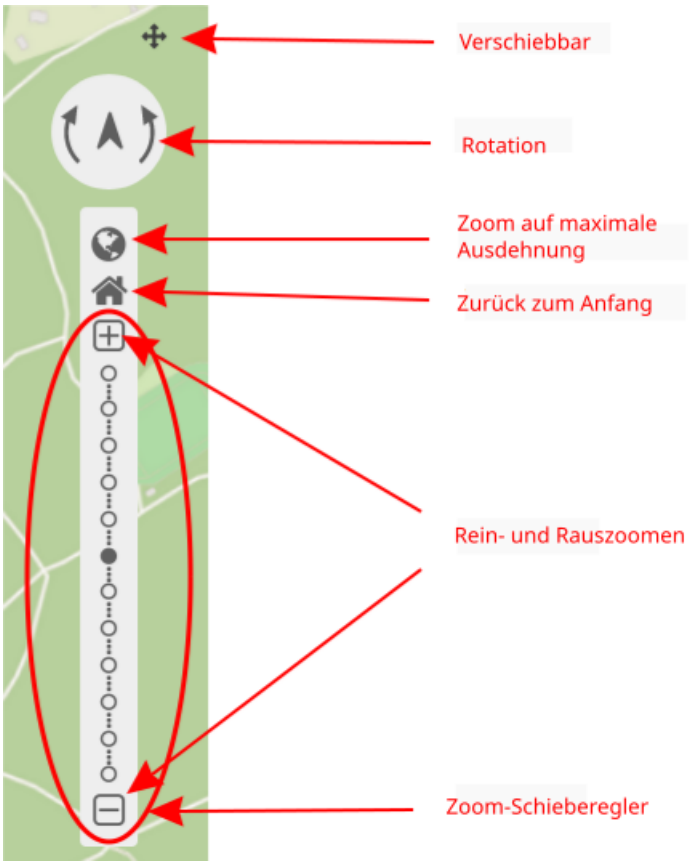
(unten links), right-top (oben rechts), right-bottom (unten rechts).

Note: Das Element muss innerhalb des Kartenbereichs konfiguriert werden.

Komponenten des Navigationswerkzeugs

Im Konfigurationsdialog können unterschiedliche Komponenten, also Funktionen des Navigationswerkszeugs, eingebunden werden.

Die Komponenten sehen in der Anwendung wie folgt aus:



- **Verschiebbar:** Wenn das Häkchen bei Verschiebbar gesetzt wurde, dann erscheint in der Anwendung ein kleines Kreuzsymbol zur Verschiebung des Navigationselements.
- **Drehen:** Mit Drehen kann die Ausrichtung der Karte verändert werden. Durch Klick auf den Nordpfeil ist es möglich, die Standardrotation (Nordausrichtung) wiederherzustellen.
- **Zoom auf maximale Ausdehnung:** Es besteht die Möglichkeit, auf den kleinsten Maßstab zu zoomen. Dieser wird innerhalb des *Kartenelements* im Kartenbereich definiert.
- **Zurück zum Anfang:** Ermöglicht das Wiederherstellen der anfänglichen Einstellungen von Maßstab, Koordinatenreferenzsystem, Rotation und Zentrierung.
- **Rein- und Rauszoomen:** Ermöglicht das Hineinzoomen, sowie das Hinauszoomen aus der Karte. Der Kartenausschnitt springt dabei immer zum jeweils nächstgrößeren bzw. -kleineren Maßstab. Je nach Definition im Kartenelement sind größere oder kleinere Zoomschritte möglich. Nutzer haben außerdem die Möglichkeit, feste Zoomstufen im Kartenelement zu deaktivieren.

- **Zoom-Schieberegler:** Die Komponente beinhaltet automatisch die Komponente *Rein- und Raus-zoomen*. Zusätzlich besteht diese Komponente aus einer Leiste, die die möglichen Zoomstufen als auswählbare Punkte anzeigt.

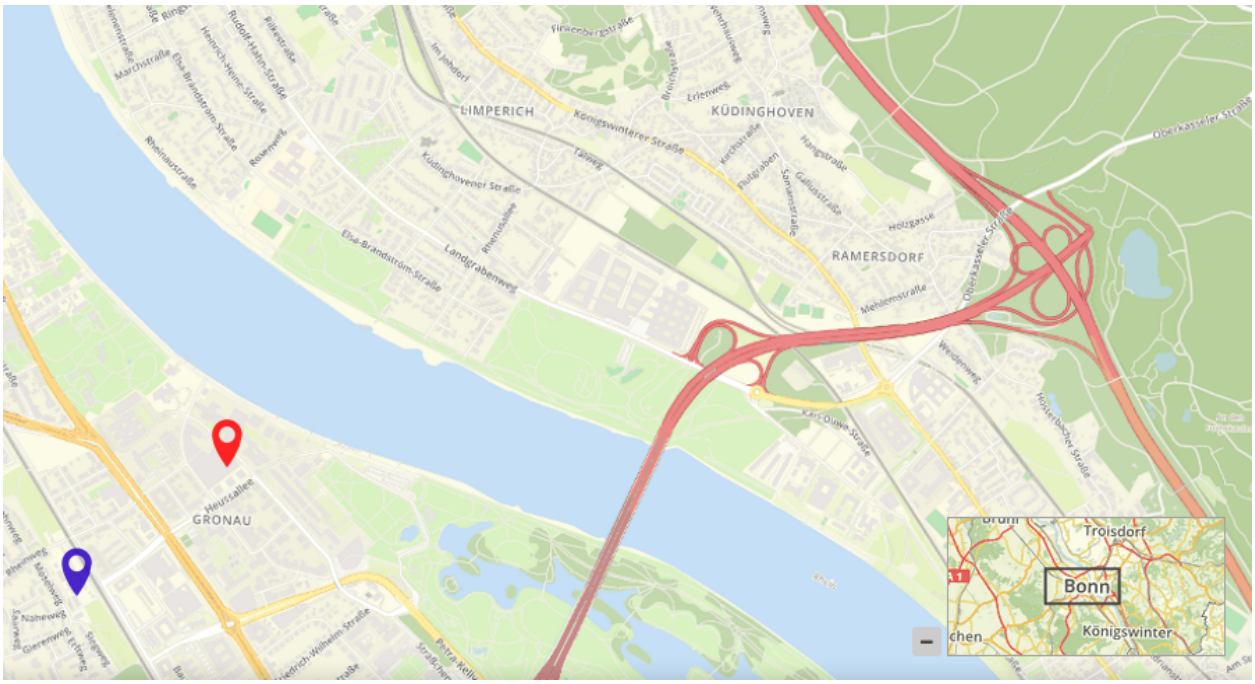
YAML-Definition

Diese Vorlage kann genutzt werden, um das Element in einer YAML-Anwendung einzubinden.

```
tooltip: 'Navigation Toolbar'     # Text des Tooltips
components: ["rotation",          # Komponenten des Navigationswerkzeugs, standardmäßig wird alles
↪selektiert
"zoom_max", "zoom_home",
"zoom_in_out", "zoom_slider"]
target: ~                         # ID des Kartenelements
anchor: left-top                  # Ausrichtung des Navigationswerkzeugs (Standard: 'left-top')
                                  # Benutzen Sie inline z.B. für die Sidebar
                                  # Optionen: 'inline', 'left-top', 'left-bottom', 'right-top', 'right-
↪bottom'
draggable: true                   # das Navigationswerkzeug ist verschiebbar oder nicht (Standard: true)
zoomHomeRestoresLayers: false     # Zurücksetzen von Dienstzuständen (Standard: false)
```

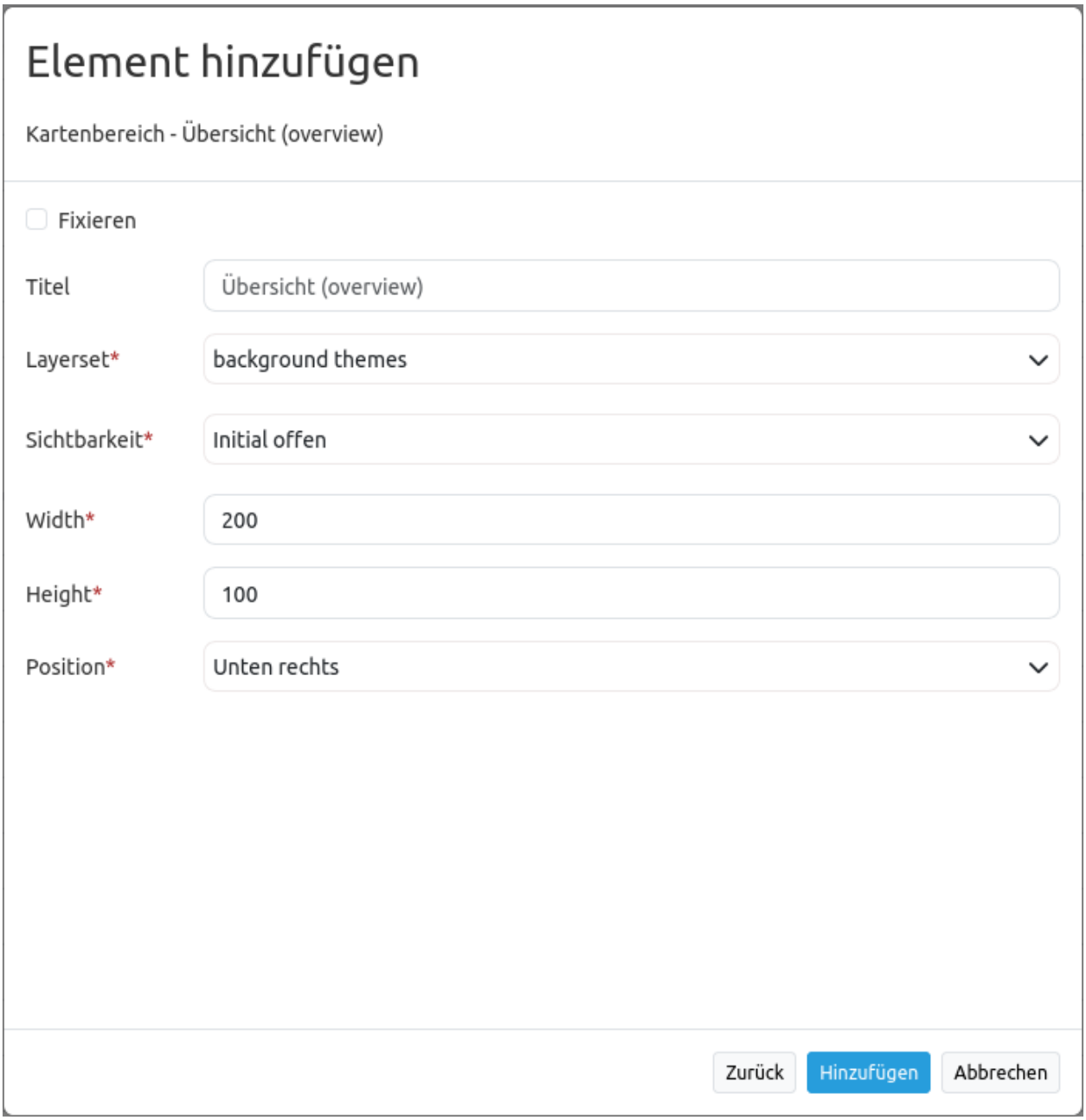
3.1.10 Übersichtskarte (Overview)

Mapbender bietet neben der Hauptkarte auch eine Übersichtskarte an. Dieses Element kann in Größe, Position und Zoom-Verhalten individuell angepasst werden. Es bezieht sich auf ein bestimmtes Layerset und die darin definierten Instanz(en).



Konfiguration

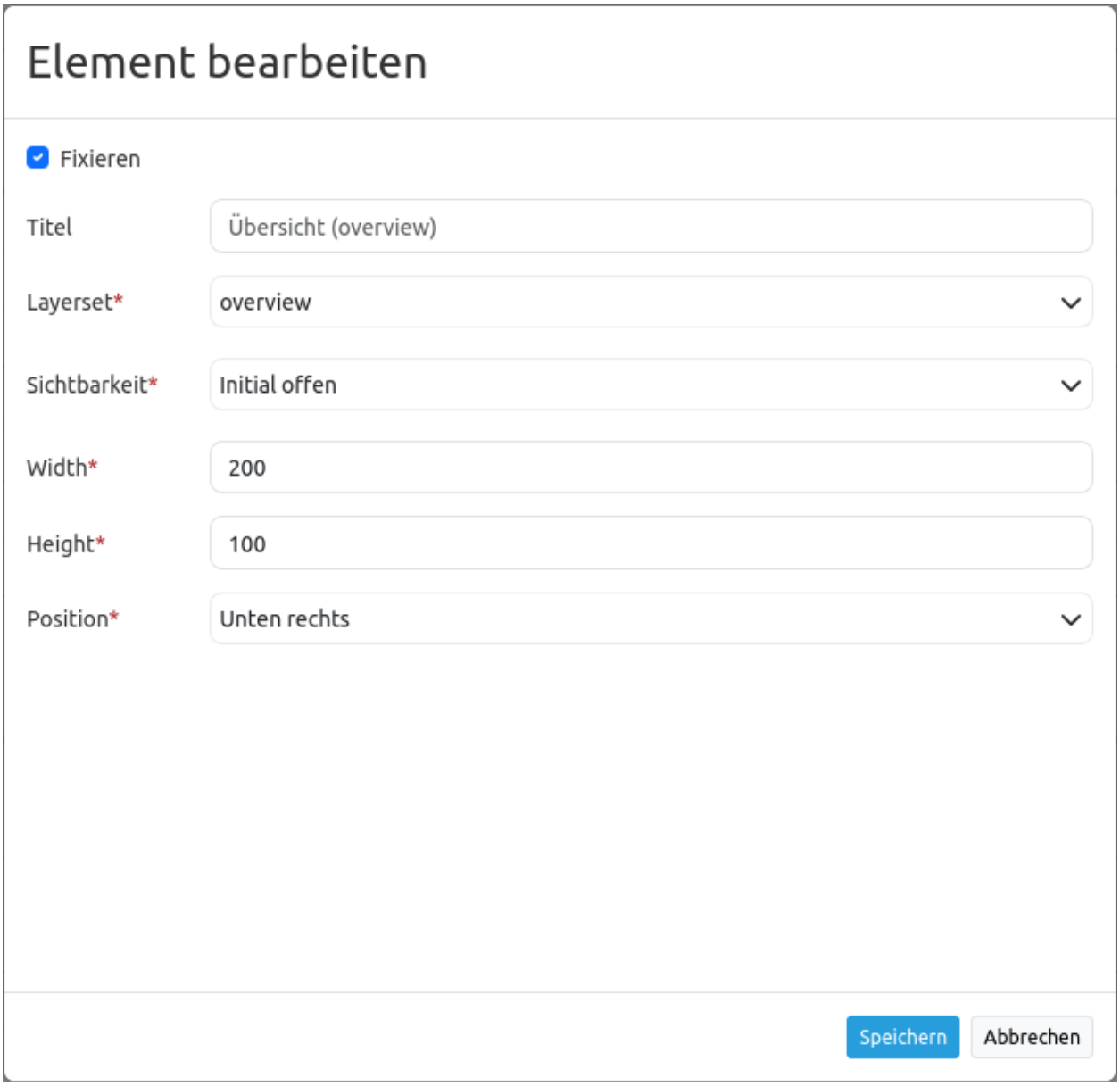
Der Konfigurationsdialog:



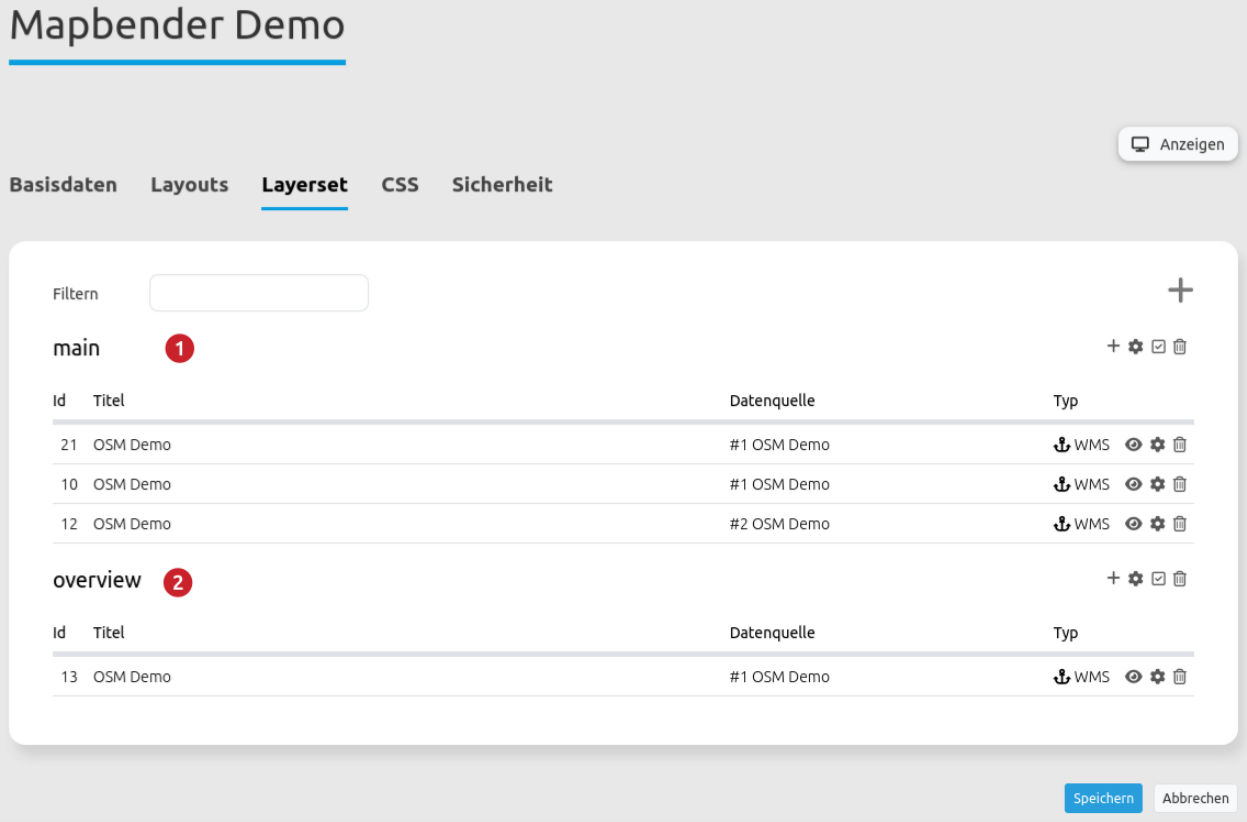
- **Fixieren:** Definiert, ob die Übersichtskarte in ihrem Maßstab fixiert sein soll (Standard: false).
- **Titel:** Titel des Elements, wird in der Layouts-Liste angezeigt.
- **Layerset:** Layerset, das im Kartenrahmen angezeigt werden soll.
- **Sichtbarkeit:** Definiert, ob der Kartenrahmen beim Start maximiert (Initial offen) oder minimiert (Initial geschlossen) oder Dauerhaft offen (kein Button zum schließen) ist (Standard: Initial offen).
- **Width:** Breite der Übersichtskarte.
- **Height:** Höhe der Übersichtskarte.

- **Position:** Position des Übersichtskartenrahmens in der Anwendung; Auswahlmöglichkeiten: oben links, unten links, oben rechts, unten rechts.

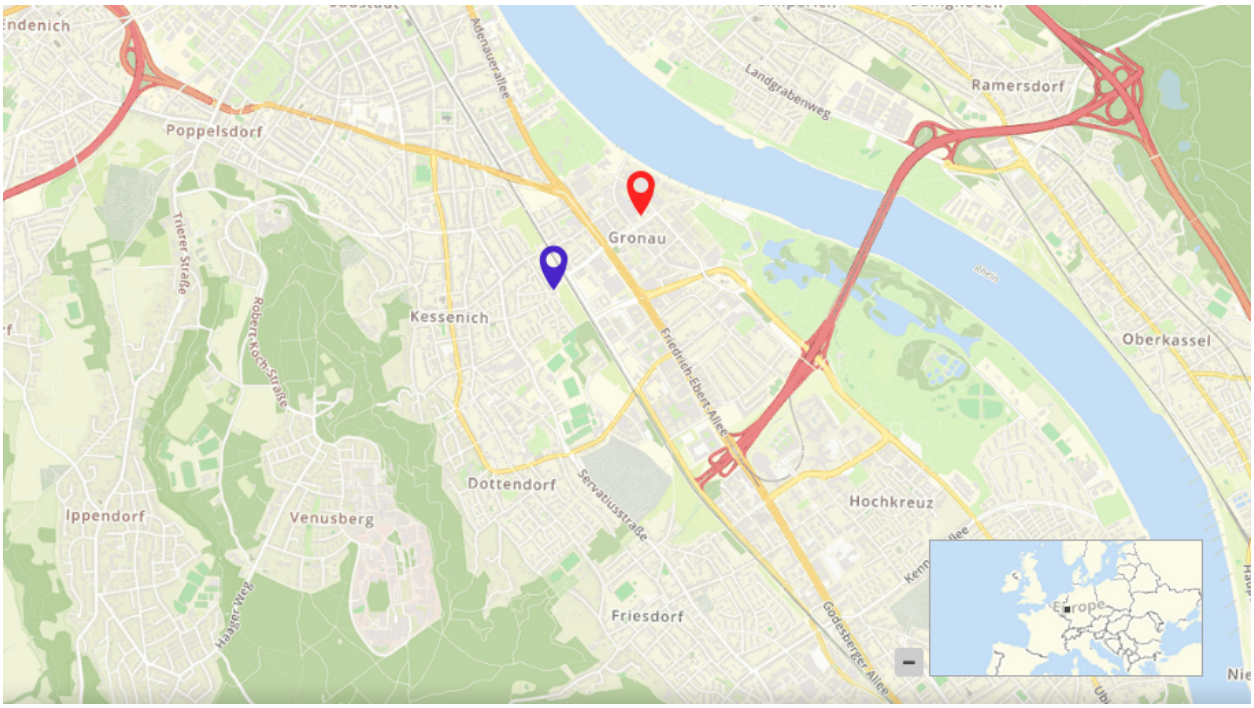
Konfigurationsbeispiele



Das Element bietet verschiedene Konfigurationsmöglichkeiten. Im Beispiel sind die Einstellungen *Sichtbarkeit Initial offen* sowie *Fixieren* gesetzt. Dadurch wird das Element beim Öffnen der Anwendung direkt angezeigt (d.h. maximiert), zusätzlich ist die Ansicht der Karte fixiert. Ist die Übersicht nicht fixiert, dann passt sich die Übersichtskarte dem entsprechenden Kartenausschnitt der Hauptkarte an, sobald diese verschoben oder ihr Maßstab verändert wird. Als Startansicht beim Öffnen der Anwendung wird der Startextent angezeigt. Das Element hat im Beispiel den Standardtitel "Übersicht (overview)". Es ist außerdem notwendig, dass die Übersichtskarte mit einem Layerset verknüpft wird. In diesem Beispiel stehen folgende Layersets zur Auswahl:

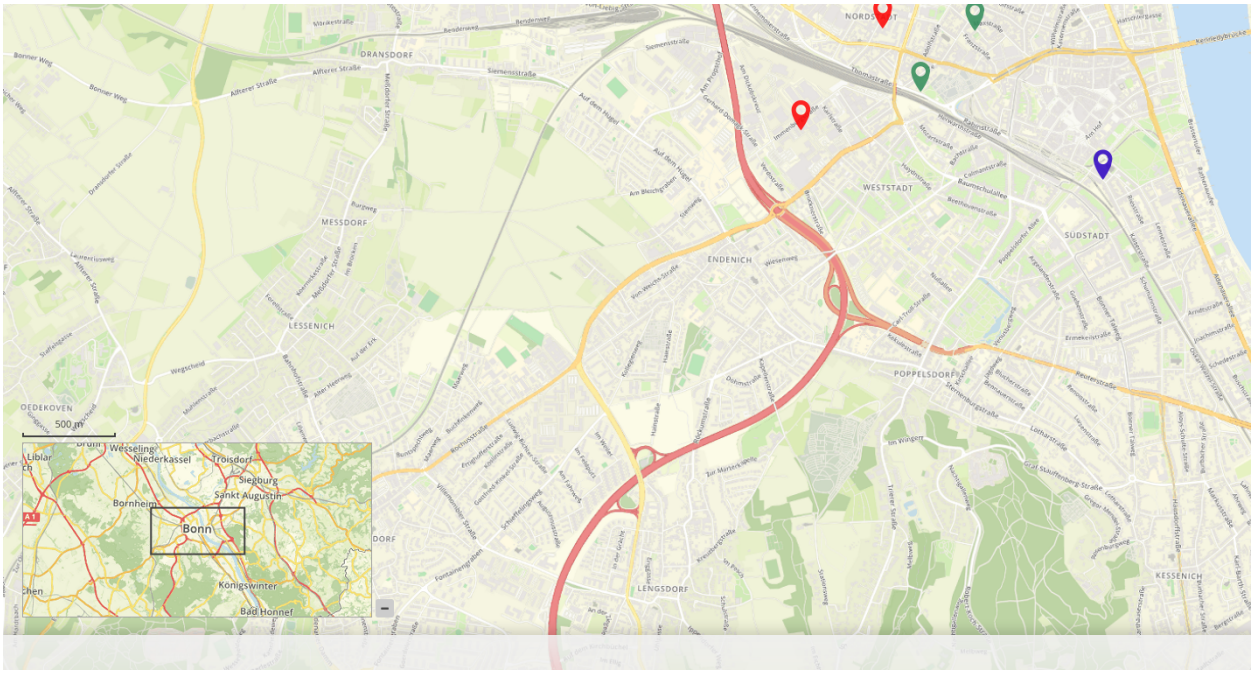


Für die Übersicht wurde das Layerset "overview" gewählt. Breite (*Width*) sowie Höhe (*Height*) des Elements entsprechen der Standardeinstellung. Die Position ist als "Unten rechts" definiert. Das Element sieht in der Anwendung folgendermaßen aus:



Ein alternatives Konfigurationsbeispiel kann die Anpassung der folgenden Parameter (*Fixieren*: nicht ak-

tiv, *Position*: unten links, *Width*: 400, *Height*: 200) sein, die eine andere Übersichtskarte (nach Einklappen der Sidepane) generiert:



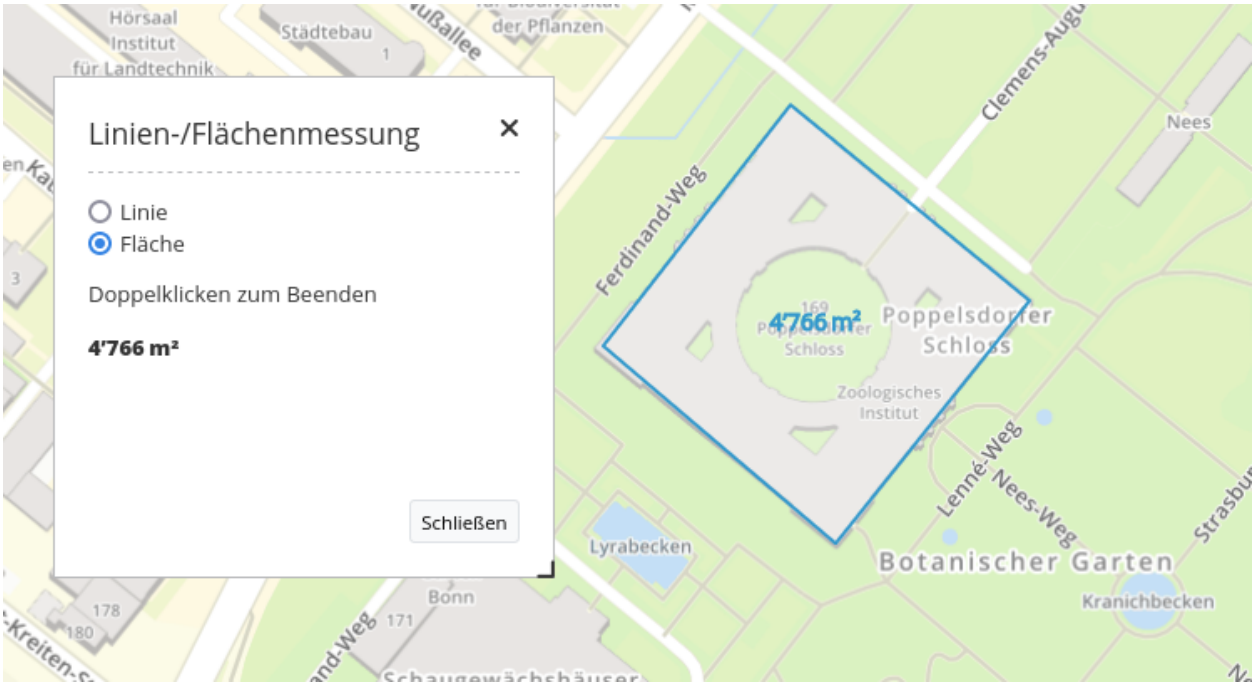
YAML-Definition

Diese Vorlage kann genutzt werden, um das Element in einer YAML-Anwendung einzubinden.

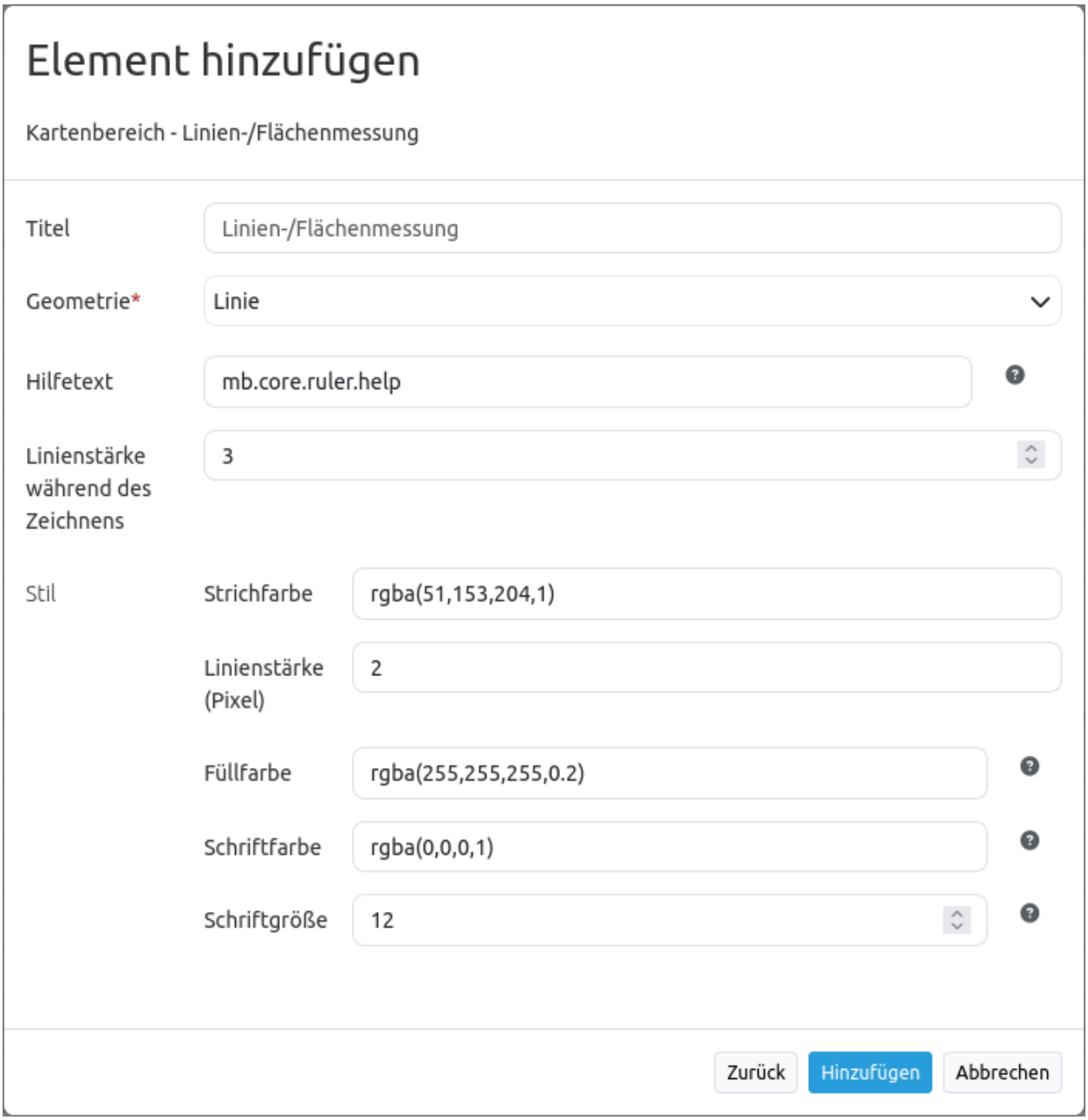
```
tooltip: 'Overview'          # Text des Tooltips
target: ~                    # ID des Kartenelements
layerset: ~                  # vorher definiertes Layerset, das angezeigt werden soll.
width: 200                   # Breite der Übersicht
height: 100                  # Höhe der Übersicht
anchor: 'right-top'          # Ausrichtung der Übersicht (Standard: right-top)
                             # Benutzen Sie inline z.B. für die Sidebar
                             # Optionen: 'inline', 'left-top', 'right-top', 'left-bottom', 'right-bottom
↪'
visibility: open             # open/closed/open-permanent - open/closed zeigt einen Button zum Öffnen/
↪Schließen (default: open), open-permanent zeigt den Button nicht an
fixed: true                  # true/false um den Übersichtsbereich zu fixieren (Standard: true)
```

3.1.11 Linien-/Flächenmessung (Line/Area Ruler)

Mit dem Messen-Werkzeug wird eine Geometrie gezeichnet, deren Länge oder Flächeninhalt berechnet wird. Das Element kann in der Sidepane oder mit einem *Button* verwendet werden. Durch die Auswahl eines Typs wird bestimmt, ob das Element Linien oder Flächen misst oder beides und so der Benutzer die Auswahl im Client treffen kann.


Linien-/Flächenmessung
Linie
Fläche
Doppelklicken zum Beenden
4'766 m²
Schließen
4766 m²
Poppelsdorfer Schloss
Botanischer Garten
Lyrabecken
Kranichbecken
Zoologisches Institut
Ferdinand-Weg
Lenné-Weg
Nees-Weg
Clemens-Aug
Nees
Hörsaal Institut für Landtechnik
Städtebau
der Pflanzen
Bonn
Schaugewächshäuser

Konfiguration



- **Titel:** Titel des Elements. Dieser wird im *Backend* in der Layouts Liste angezeigt. In der Anwendung selbst wird der Titel im Messfenster angezeigt.
- **Geometrie:** Typ des Elements, entweder 'Linie', 'Fläche' oder 'vom Nutzer auswählbar'. Pflichtfeld.
- **Hilfetext:** Definition eines individuellen Hilfetextes. Der Standardwert mb.core.ruler.help ist in der Mapbender Übersetzungsdatei definiert und enthält auf Deutsch den Text "Doppelklicken zum Beenden". Der Text variiert je nach Anzeigesprache des Browsers.
- **Linienstärke während des Zeichnens:** Zahlenwert in Pixel, der die Strichstärke während des Zeichnens definiert.
- **Strichfarbe:** RGBA-Wert, der die Strichfarbe definiert. Kann nach einem Klick in die Eingabemaske

über einen Farbwähler geändert werden.

- **Linienstärke (Pixel):** Zahlenwert in Pixel, der die Strichstärke der gemessenen Geometrie definiert.
- **Füllfarbe:** RGBA-Wert, der die Füllfarbe einer Fläche definiert. Kann nach einem Klick in die Eingabemaske über einen Farbwähler geändert werden. Wird als Geometrie *Linie* gewählt, bleibt die Option ohne Effekt.
- **Schriftfarbe:** RGBA-Wert, der die Schriftfarbe des in der Geometrie angezeigten numerischen Flächeninhalts ausgibt. Kann nach einem Klick in die Eingabemaske über einen Farbwähler geändert werden. Wird als Geometrie *Linie* gewählt, bleibt die Option ohne Effekt.
- **Schriftgröße:** Zahlenwert, der die Schriftgröße der in der Geometrie angezeigten Fläche definiert. Wird als Geometrie *Linie* gewählt, bleibt die Option ohne Effekt.

YAML-Definition

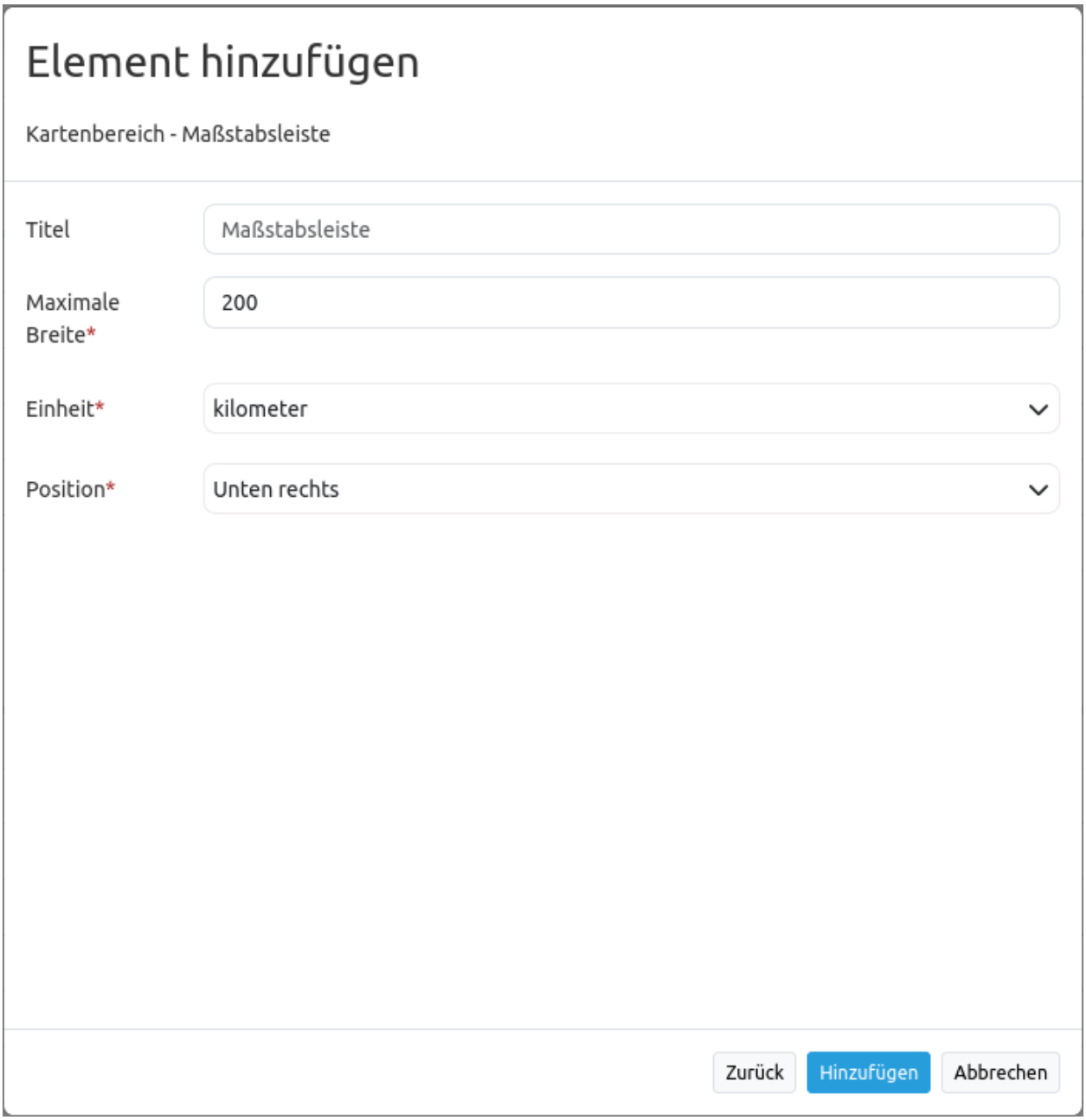
Diese Vorlage kann genutzt werden, um das Element in einer YAML-Applikation einzubinden.

```
title: mb.core.ruler.class.title          # 'Linien-/Flächenmessung'
class: Mapbender\CoreBundle\Element\Ruler # Element-Klasse
target: map                               # ID des Kartenelements, z.B. 'map'
type: line                                # Wählen Sie Typ 'line', 'area' oder 'both'
strokeColor: 'rgba(16, 101, 93, 0.8)'     # Angabe der Strichfarbe (für Linien und Flächen)
fillColor: 'rgba(100, 100, 100, 0.5)'     # Angabe der Füllfarbe (nur Flächenmessung)
fontColor: 'rgba(0,0,0,1)'                # Angabe der Schriftarbe (nur Flächenmessung)
fontSize: 14                              # Angabe der Schriftgröße (nur Flächenmessung)
strokeWidth: 4                            # Angabe der Linienstärke, in px (für Linien und Flächen)
strokeWidthWhileDrawing: 3                # Angabe der Linienstärke während des Zeichnens, in px (für
↪Linien und Flächen)
```

3.1.12 Maßstabsleiste (Scale bar)

Die Maßstabsleiste stellt den aktuellen Maßstab graphisch als Linie dar:

Konfiguration



- **Titel:** Titel des Elements. Dieser wird unter dem Reiter *Layouts* angezeigt.
- **Maximale Breite:** Maximale Breite der Maßstabsleiste (Pflichtfeld; Standard: 200px).
- **Einheit:** Die wählbaren Einheiten der Maßstabsleiste:'kilometer' oder 'miles' (Pflichtfeld; Standard: kilometer).
- **Position:** Ausrichtung der Maßstabsleiste (Pflichtfeld; Standard: Unten rechts).

YAML-Definition

Diese Vorlage kann genutzt werden, um das Element in einer YAML-Anwendung einzubinden.

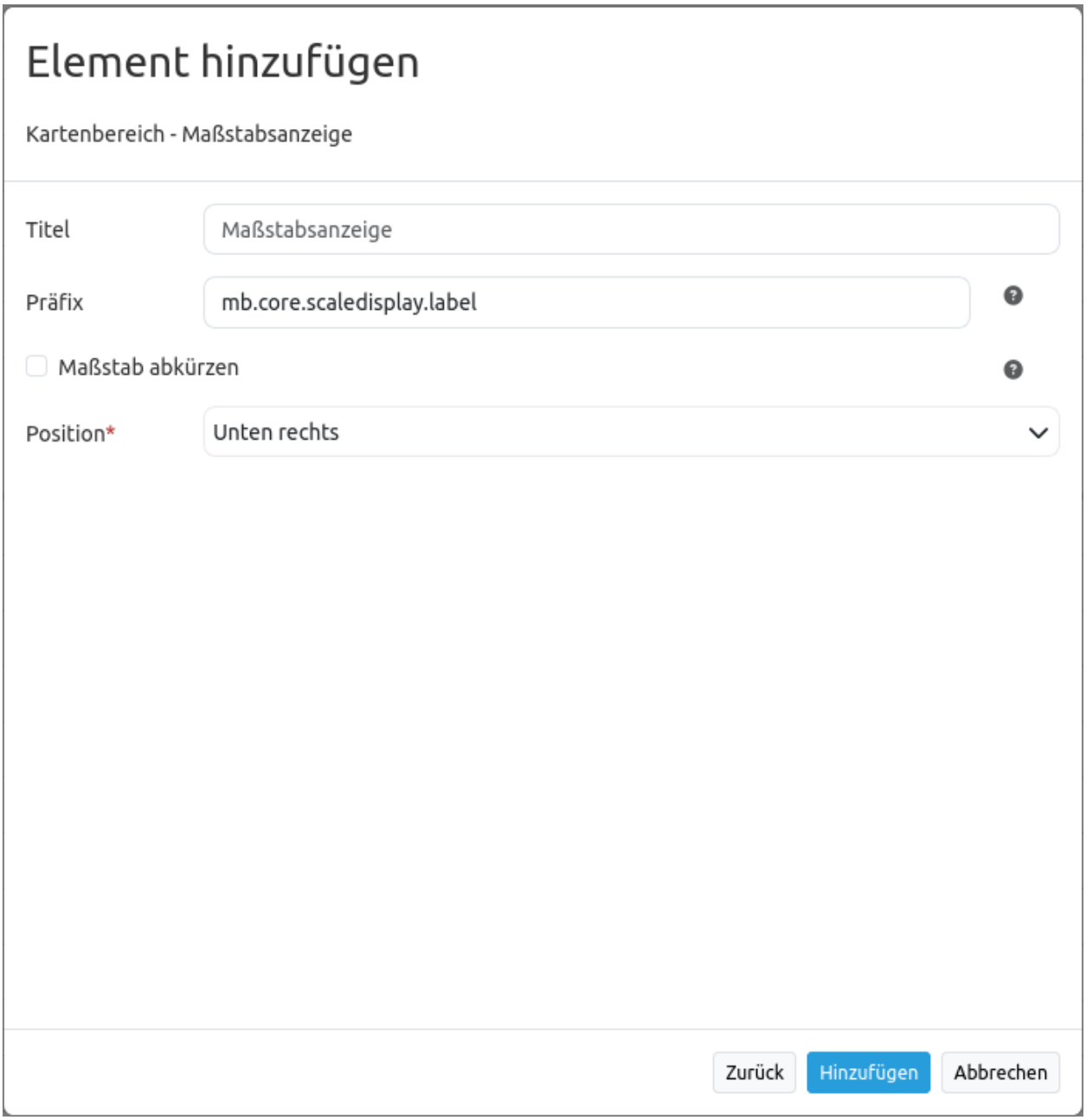
```
tooltip: 'Scale Bar'              # Text des Tooltips
target: ~                         # ID des Kartenelements
anchor: 'inline'/'left-top'/      # Ausrichtung des Maßstabsbalkens (Standard: 'right-bottom')
  'left-bottom'/'right-top'/      # Benutzen Sie inline z.B. für die Sidebar
  'right-bottom'
position: ['10px', '10px']        # Position des Maßstabsbalkens (Standard: x=20px, y=20px)
maxWidth: 200                     # Maximale Breite des Maßstabsbalkens (Standard: 200)
units: ['km']                     # Einheiten des Maßstabsbalkens, 'kilometer' und/oder 'miles' (ml),
↪(Standard: ['km'])
```

Pflichtfeld;

3.1.13 Maßstabsanzeige (Scale display)

Die Maßstabsanzeige zeigt den aktuellen Maßstab numerisch an (z.B. als 1:7.500 oder 1:10K).

Konfiguration



- **Titel:** Titel des Elements. Dieser wird in der Layouts Liste angezeigt und ermöglicht, mehrere Button-Elemente voneinander zu unterscheiden.
- **Präfix:** Anpassbare Bezeichnung, die vor der Maßstabsangabe steht. Der Standard `mb.core.scaledisplay.label` wird als Maßstab in der Sprache des Benutzers gerendert.
- **Maßstab abkürzen:** Falls aktiviert, werden Maßstabszahlen ab 10.000 nicht ausgeschrieben, sondern mit nachgestelltem `K` oder `M` versehen, z.B. 10K für 10.000 (Standard: false).
- **Position:** Position des Elements (Pflichtfeld, Standard: 'Unten rechts').

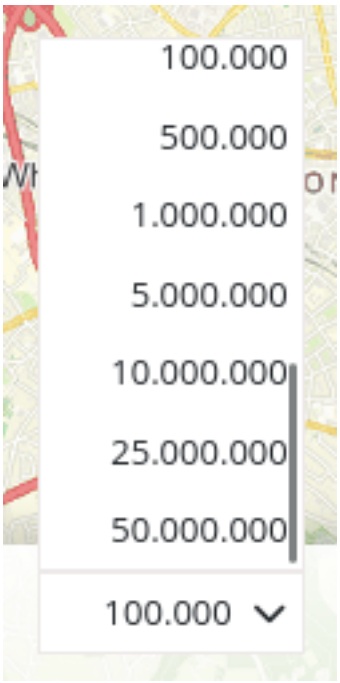
YAML-Definition

Diese Vorlage kann genutzt werden, um das Element in einer YAML-Anwendung einzubinden.

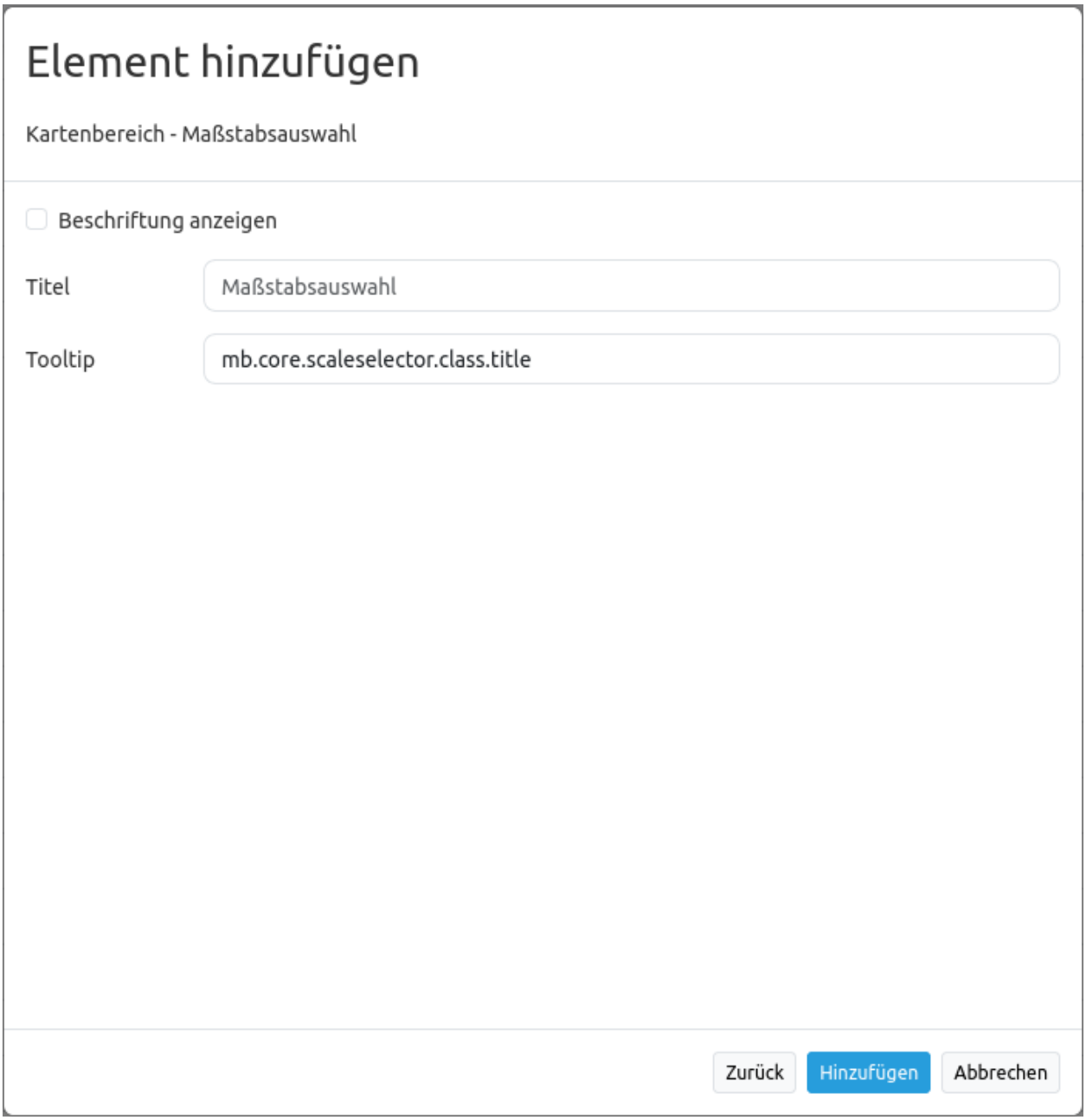
```
tooltip: 'ScaleDisplay'          # Text des Tooltips
target: ~                        # ID des Kartenelements
anchor: 'inline'/'left-top'/     # Ausrichtung des ScaleDisplay-Elements (Standard: 'right-bottom')
  'left-bottom'/'right-top'/
  'right-bottom'
scalePrefix: Scale               # Bezeichnung, die vor der Maßstabsangabe steht, z.B. "Maßstab"
unitPrefix: true                 # aktiviert Darstellung der Maßstabszahl mit Präfix (Standard: false).
```

3.1.14 Maßstabsauswahl (Scale selector)

Dieses Element zeigt eine Auswahlbox mit Maßstäben an. Der Kartenmaßstab ändert sich, wenn ein neuer Wert der Auswahlbox ausgewählt wird. Zudem ändert sich der Wert der Anzeige, wenn in die Karte hinein- oder herausgezoomt wird. Die Auswahlbox zeigt nur Maßstäbe an, die vorher auch für das Map Element definiert wurden.



Konfiguration



- **Beschriftung anzeigen:** Zeigt die Beschriftung der Maßstabsauswahl an (Standard: false).
- **Titel:** Titel des Elements. Dieser wird unter dem Reiter *Layouts* angezeigt und ermöglicht, mehrere Button-Elemente voneinander zu unterscheiden. Der Titel wird außerdem neben der Auswahlbox angezeigt, wenn die Option "Beschriftung anzeigen" aktiviert ist.
- **Tooltip:** Text, der angezeigt wird, wenn der Mauszeiger eine längere Zeit über dem Element verweilt.

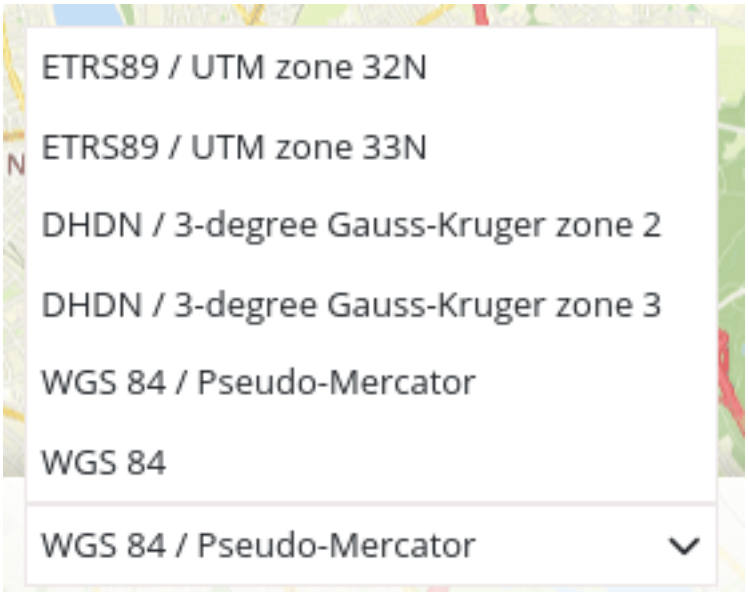
YAML-Definition

Diese Vorlage kann genutzt werden, um das Element in einer YAML-Anwendung einzubinden.

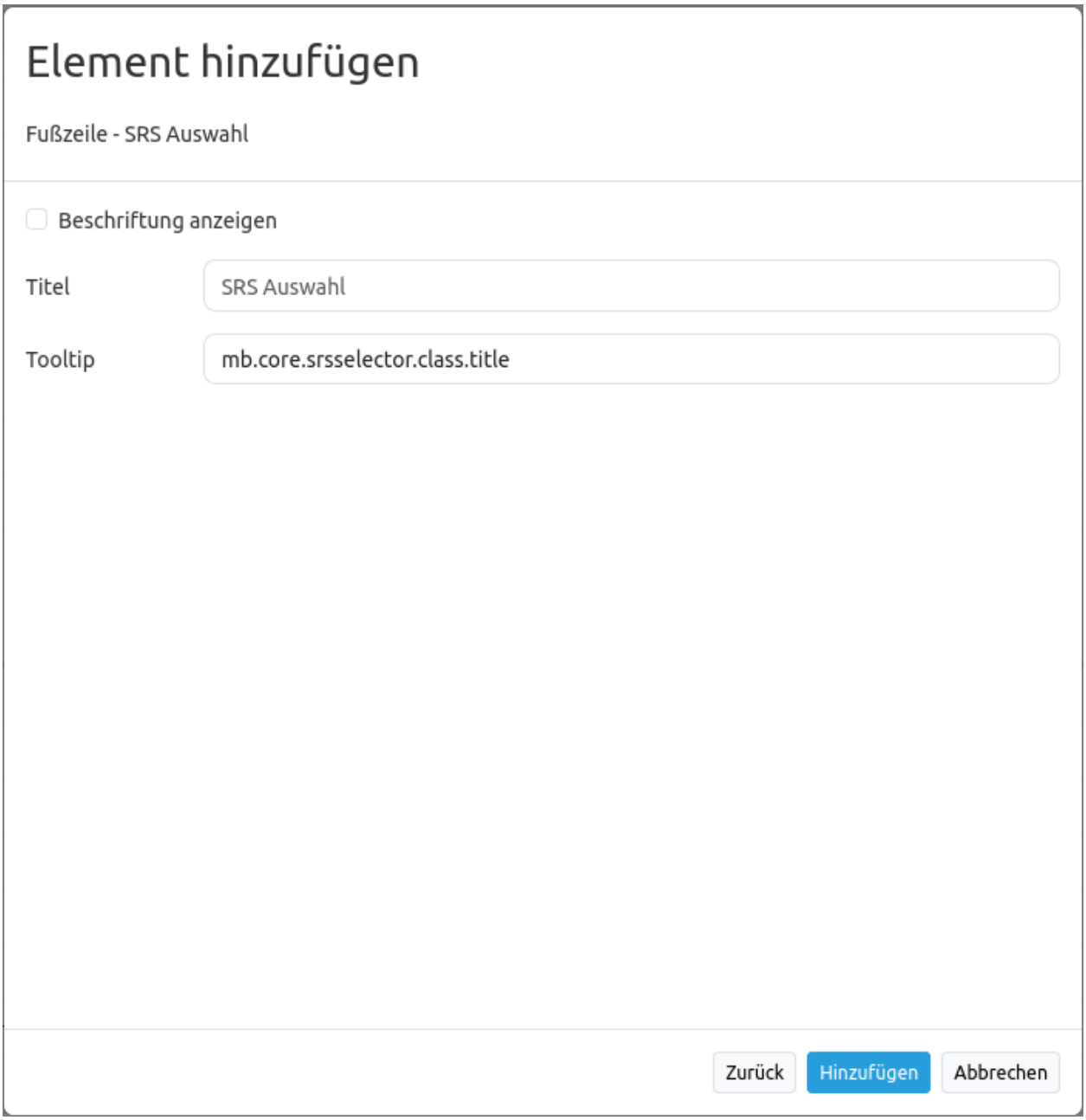
```
tooltip: "Maßstabsauswahl"  # Text des Tooltips
target: ~          # ID des Kartenelements
label: false       # false/true, um die Maßstabsauswahl zu beschriften (Standard: false).
```

3.1.15 SRS Auswahl (SRS Selector)

Mit der SRS Auswahl kann das räumliche Referenzsystem (SRS) der Karte geändert werden. Nach der Konfiguration stehen die Koordinatensysteme der Karte in einer Selectbox zur Auswahl, die vorher im *Kartenelement* definiert wurden.



Konfiguration



- **Beschriftung anzeigen:** Beschriftet die SRS Auswahl (Standard: false).
- **Titel:** Titel des Elements. Dieser wird in der Layouts-Liste angezeigt und ermöglicht die Unterscheidung mehrerer Button-Elemente voneinander. Der Titel wird außerdem neben dem SRS Selector angezeigt, wenn "Beschriftung anzeigen" aktiviert ist.
- **Tooltip:** Text, der angezeigt wird, wenn der Mauszeiger eine längere Zeit über dem Element verweilt.

YAML-Definition

Diese Vorlage kann genutzt werden, um das Element in einer YAML-Anwendung einzubinden.

```
tooltip: 'SRS Selector'  # Text des Tooltips
label: false             # false/true, um den SRS Selector zu beschriften (Standard: false).
target: ~                # ID des Kartenelements
```

3.2 Editierung

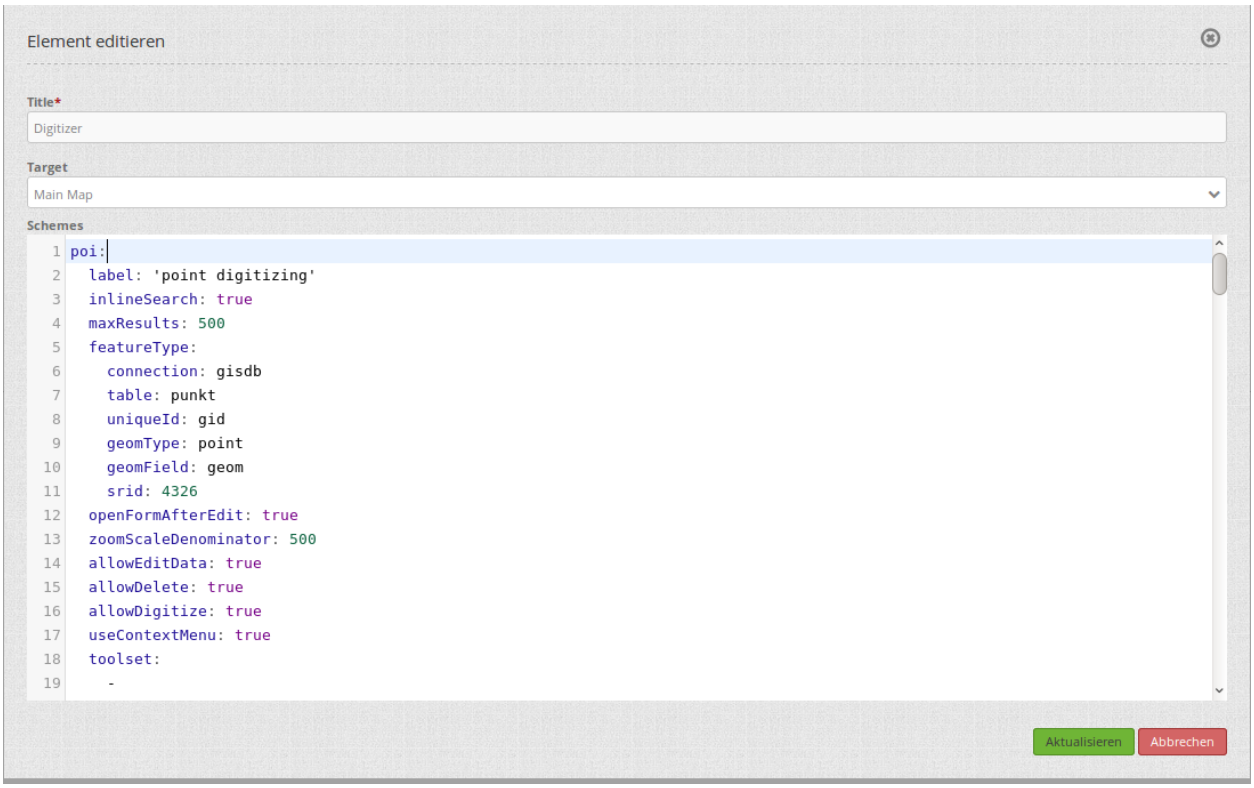
3.2.1 Digitizer

Der Digitizer erlaubt das Erstellen und Bearbeiten von Punkten, Linien und Flächen in der Karte. Im Unterschied zum Sketch-Element werden die Geometrien hierbei in einer Tabelle in einer Datenbank hinterlegt. Bisher unterstützt Mapbender PostgreSQL als Datenquelle. SpatiaLite und Oracle sind experimentell verfügbar. Die Entwicklung wurde so durchgeführt, dass die Erfassung auch auf andere Datenquellen wie z.B. OGC WFS erweitert werden kann.

Um das Digitizer-Element nutzen zu können, muss eine YAML-Definition aufgebaut werden:

Einrichtung des Digitizers

Das Element kann nur in der Sidepane eingebettet werden.



- **Title:** Titel des Elements. Dieser wird in der Layouts Liste angezeigt und ermöglicht, mehrere Elemente voneinander zu unterscheiden.

- **Target:** Zielelement (Titel(ID)) der Karte.
- **Schemes:** YAML-Definition für das Element "digitizer"

Der Digitizer benötigt einen Zugriff auf die Datenbank, in der die zu editierenden Tabellen liegen. Sie müssen dazu einen Datenbankzugriff konfigurieren. Mehr zu diesem Thema finden Sie unter *YAML Konfiguration (Konfigurations- und Anwendungsdateien)*.

Die Definition des Digitizers wird in einer YAML-Syntax durchgeführt. Hier definieren Sie die Datenbankverbindung, die editierbaren Felder, das Formular für die Anzeige und andere Verhaltensweisen.

Hint: Bei fehlerhaften Angaben zur Datenbank, Feldern und Formularfehlern erscheinen Fehlermeldungen. Bei Produktivumgebungen erscheint eine allgemeine Fehlermeldung. Falls Sie die detaillierte Fehlermeldung sehen möchten, sollten Sie die Anwendung als Entwicklungsumgebung aufrufen. Weitergehende Informationen unter *Starten von Mapbender als Produktivumgebung*.

YAML-Definition für das Element digitizer in der Textarea unter schemes

In dem folgenden YAML-Block ist die beispielhafte Definition für drei Erfassungsoberflächen enthalten. Kopieren Sie den folgenden Block in Ihr Digitizer-Element, um die Erfassung von Punkten, Linien und Polygonen zu testen. Vorher müssen Sie die Datenbankverbindung und die drei Demo-Tabellen anlegen. Die SQL-Befehle für das Anlegen der Tabellen finden Sie weiter unten. Der Funktionsumfang der eingebauten Features und weitere Funktionen werden nach diesem Beispielaufbau genauer erläutert.

```
poi:
    label: point digitizing
    inlineSearch: true
    featureType:
        connection: search_db
        table: poi
        uniqueId: gid
        geomType: point
        geomField: geom
        srid: 4326
    openFormAfterEdit: true
    zoomScaleDenominator: 500
    allowEdit: true
    allowDelete: true
    allowDigitize: true
    useContextMenu: true
    toolset:
        - type: drawPoint
        - type: moveFeature
    popup:
        title: point test suite
        width: 500px
    searchType: currentExtent
    tableFields:
        gid: {label: Nr. , width: 20%}
        name: {label: Name , width: 80%}
    styles:
        default:
            strokeWidth: 2
            strokeColor: '#0e6a9e'
            fillColor: '#1289CD'
```

(continues on next page)

(continued from previous page)

```
            fillOpacity: 1
            fillWidth: 2
            pointRadius: 10
        select:
            strokeWidth: 3
            strokeColor: '#0e6a9e'
            fillOpacity: 0.7
            pointRadius: 10
    formItems:
       - type: tabs
         children:
           - type: form
             title: Basic information
             css: {padding: 10px}
             children:
                 - type: label
                   title: Welcome to the digitize demo. Try the new Mapbender feature!
                 - type: input
                   title: Name
                   mandatory: true
                   name: name
                   mandatoryText: Please give a name to the poi.
                   infoText: "Help: Please give a name to the new object."
                 - type: input
                   title: Title
                   mandatory: false
                   name: title
                   mandatoryText: Please give a title to the poi.
                 - type: textArea
                   name: abstract
                   title: Abstract
                   placeholder: 'please edit this field'
                 - type: select
                   title: Type
                   name: type
                   options: {A: A, B: B, C: C, D: D, E: E}
                 - type: breakLine
           - type: form
             title: Personal information
             css: {padding: 10px}
             children:
                 - type: label
                   title: Please give us some information about yourself.
                 - type: fieldSet
                   children:
                       - type: input
                         title: Firstname
                         name: firstname
                         css: {width: 30%}
                       - type: input
                         title: Lastname
                         name: lastname
                         css: {width: 30%}
                       - type: input
                         title: E-Mail
                         name: email
```

(continues on next page)

(continued from previous page)

```
                        css: {width: 40%}
                - type: select
                  multiple: false
                  title: Interests
                  name: interests
                  options: {maps: maps, reading: reading, swimming: swimming, dancing: dancing, beer:
↪beer, flowers: flowers}
                - type: date
                  title: favorite Date
                  name: date_favorite
                  mandatory: true
                  css: {width: 25%}
                - type: breakLine
                - type: breakLine
                - type: checkbox
                  name: public
                  value: true
                  title: public (this new object is public)
line:
    label: line digitizing
    inlineSearch: true
    featureType:
        connection: search_db
        table: lines
        uniqueId: gid
        geomType: line
        geomField: geom
        srid: 4326
    openFormAfterEdit: true
    allowDelete: true
    useContextMenu: true
    toolset:
        - type: drawLine
        - type: modifyFeature
        - type: moveFeature
    popup:
        title: line test suite
        width: 500px
    searchType: currentExtent
    tableFields:
        gid: {label: Nr. , width: 20%}
        name: {label: Name , width: 80%}
    styles:
        default:
            strokeWidth: 2
            strokeColor: '#0e6a9e'
            fillColor: '#1289CD'
            fillOpacity: 1
            fillWidth: 2
            pointRadius: 10
        select:
            strokeWidth: 3
            strokeColor: '#0e6a9e'
            fillOpacity: 0.7
            pointRadius: 10
    formItems:
```

(continues on next page)

(continued from previous page)

```
      - type: form
        title: Basic information
        css: {padding: 10px}
        children:
            - type: label
              title: Welcome to the digitize demo. Try the new Mapbender feature!
            - type: input
              title: Name
              name: name
              mandatory: true
              mandatoryText: Please give a name to the new object.
              infoText: "Help: Please give a name to the new object."
            - type: select
              title: Type
              name: type
              options: {A: A, B: B, C: C, D: D, E: E}
polygon:
    label: polygon digitizing
    inlineSearch: true
    featureType:
        connection: search_db
        table: polygons
        uniqueId: gid
        geomType: polygon
        geomField: geom
        srid: 4326
    openFormAfterEdit: true
    allowDelete: false
    useContextMenu: true
    toolset:
        - type: drawPolygon
        - type: drawRectangle
        - type: drawDonut
        - type: drawEllipse
        - type: drawCircle
        - type: modifyFeature
        - type: moveFeature
    popup:
        title: polygon test suite
        width: 500px
    searchType: currentExtent
    tableFields:
        gid: {label: Nr. , width: 20%}
        name: {label: Name , width: 80%}
    styles:
        default:
            strokeWidth: 2
            strokeColor: '#0e6a9e'
            fillColor: '#1289CD'
            fillOpacity: 1
            fillWidth: 2
            pointRadius: 10
        select:
            strokeWidth: 3
            strokeColor: '#0e6a9e'
            fillOpacity: 0.7
```

(continues on next page)

(continued from previous page)

```
            pointRadius: 10
    formItems:
       - type: form
         title: Basic information
         css: {padding: 10px}
         children:
             - type: label
               title: Welcome to the digitize demo. Try the new Mapbender feature!
             - type: input
               title: Name
               mandatory: true
               name: name
               mandatoryText: Please give a name to the new object.
               infoText: "Help: Please give a name to the new object."
             - type: select
               title: Type
               name: type
               options: {A: A, B: B, C: C, D: D, E: E}
```

SQL für die Demo-Tabellen

Die folgenden SQL-Befehle müssen in Ihrer Datenbank ausgeführt werden. Sie legen drei Demo-Tabellen an, damit mit der oben gezeigte YAML-Definition die einzelnen Funktionen getestet werden können. Die PostGIS Extension muss aktiviert sein.

```
create table public.poi (
    gid serial PRIMARY KEY,
    name varchar,
    type varchar,
    abstract varchar,
    public boolean,
    date_favorite date,
    title varchar,
    firstname varchar,
    lastname varchar,
    email varchar,
    interests varchar,
    user_name varchar,
    group_name varchar,
    modification_date date,
    my_type varchar,
    file_reference varchar,
    x float,
    y float,
    city varchar,
    geom geometry(point,4326)
);
```

```
create table public.lines (
    gid serial PRIMARY KEY,
    name varchar,
    type varchar,
    abstract varchar,
    public boolean,
```

(continues on next page)

(continued from previous page)

```
    date_favorite date,
    title varchar,
    firstname varchar,
    lastname varchar,
    email varchar,
    interests varchar,
    length float,
    category varchar,
    user_name varchar,
    group_name varchar,
    modification_date date,
    my_type varchar,
    file_reference varchar,
    x float,
    y float,
    city varchar,
    geom geometry(linestring,4326)
);
```

```
create table public.polygons (
    gid serial PRIMARY KEY,
    name varchar,
    type varchar,
    abstract varchar,
    public boolean,
    date_favorite date,
    title varchar,
    firstname varchar,
    lastname varchar,
    email varchar,
    interests varchar,
    area float,
    category varchar,
    user_name varchar,
    group_name varchar,
    modification_date date,
    my_type varchar,
    file_reference varchar,
    x float,
    y float,
    city varchar,
    geom geometry(polygon,4326)
);
```

Konfiguration

Im Folgenden werden die einzelnen Bestandteile des Digitizers erklärt, die die Grundstruktur ausmachen und die in dem Formular eingebettet werden können.

Basisdefinition

Eine Basisdefinition, hier am Beispiel der poi, sieht folgendermaßen aus:

```
poi:
    label: point digitizing
    minScale: 5000
    featureType:
        connection: search_db
        table: poi
        uniqueId: gid
        geomType: point
        geomField: geom
        srid: 4326
        filter: interests = 'maps'
    openFormAfterEdit: true
    zoomScaleDenominator: 500
    allowEdit: true
    allowDelete: true
    allowDigitize: true
    [...]
    popup:
        [...]
```

Die möglichen Optionen sind:

- **label:** Beschriftung mit dem Namen der Erfassungsoberfläche
- **minScale:** Minimaler Maßstab, ab dem die Features in der Karte angezeigt werden (z.B. minscale: 5000 = Anzeige ab einem Maßstab über 1:5000, beim rauszoomen)
- **featureType: Verbindung zur Datenbank**
  - connection: Name der Datenbank-Verbindung aus der parameters/doctrine.yaml
  - table: Name der Tabelle, in der das FeatureType gespeichert wird
  - uniqueId: Name der Spalte mit dem eindeutigen Identifier (Standard bei Leerwert: [id])
  - geomType: Geometrietyp
  - geomField: Attributspalte, in der die Geometrie liegt.
  - srid: Koordinatensystem im EPSG-Code
  - filter: Datenfilter über Werte in einer definierten Spalte, z.B. filter: interests = 'maps'
- **openFormAfterEdit:** Nach der Erfassung einer Geometrie öffnet sich das Erfassungsformular (Standard: true).
- **zoomScaleDenominator:** Zoomstufen, die für das Zoomen auf das Objekt gewählt wird (Standard: 100).
- **allowEdit:** Entscheidet, ob Formulardaten editiert und Geometrien gespeichert werden dürfen [true/false]. Ob die Geometrie bearbeitet werden darf, entscheidet allowDigitize.

- **allowDigitize:** Entscheidet, ob Geometrien editiert werden dürfen, z.B. verschieben der kompletten Geometrie, hinzufügen von Stützpunkten zu Polygonen etc. [true/false]. Falls true, erscheinen die Digitalisierungs-Schaltflächen (neuer Punkt, verschieben, etc.). Hat nur einen Effekt, wenn auch allowEdit auf true gesetzt ist.
- **allowDelete:** Daten dürfen gelöscht werden. [true/false]. Es erscheint eine Löschen Schaltfläche.
- **useContextMenu:** Anzeige des Kontextmenü eines Features durch Rechtsklick auf der Karte. [true/false]
- **allowCancelButton:** Zeigt die Abbrechen Schaltfläche. [true/false]. Siehe *Speichern, Löschen, Abbrechen*.
- **allowDeleteByCancelNewGeometry:** Wenn auf true gestellt: Beim Neuanlegen eines Features verhält sich der Abbrechen Knopf wie der Löschen Knopf: Das Feature wird aus der Karte und der Tabelle entfernt. Dies gilt nicht bei dem Ändern eines vorhandenen Features. [true/false]
- **displayOnInactive:** Der aktuellen FeatureType wird weiterhin auf der Karte angezeigt, auch wenn der Digitizer in der Sidepane (Accordion, Tabs) nicht mehr aktiviert ist. [true/false]. Die Option ist, wenn angeschaltet, ein wenig tricky, da auch die einzelnen Digitizer Events noch aktiviert sind, für erfahrene Anwendern aber durchaus hilfreich.
- **allowLocate:** Navigation zu einem Feature hin über die Bedienung mit der Tabs-Taste, sinvoll für die Bedienung ohne Maus. [true/false]. Es erscheint eine extra "ZoomTo" Schaltfläche zu jeden Feature.

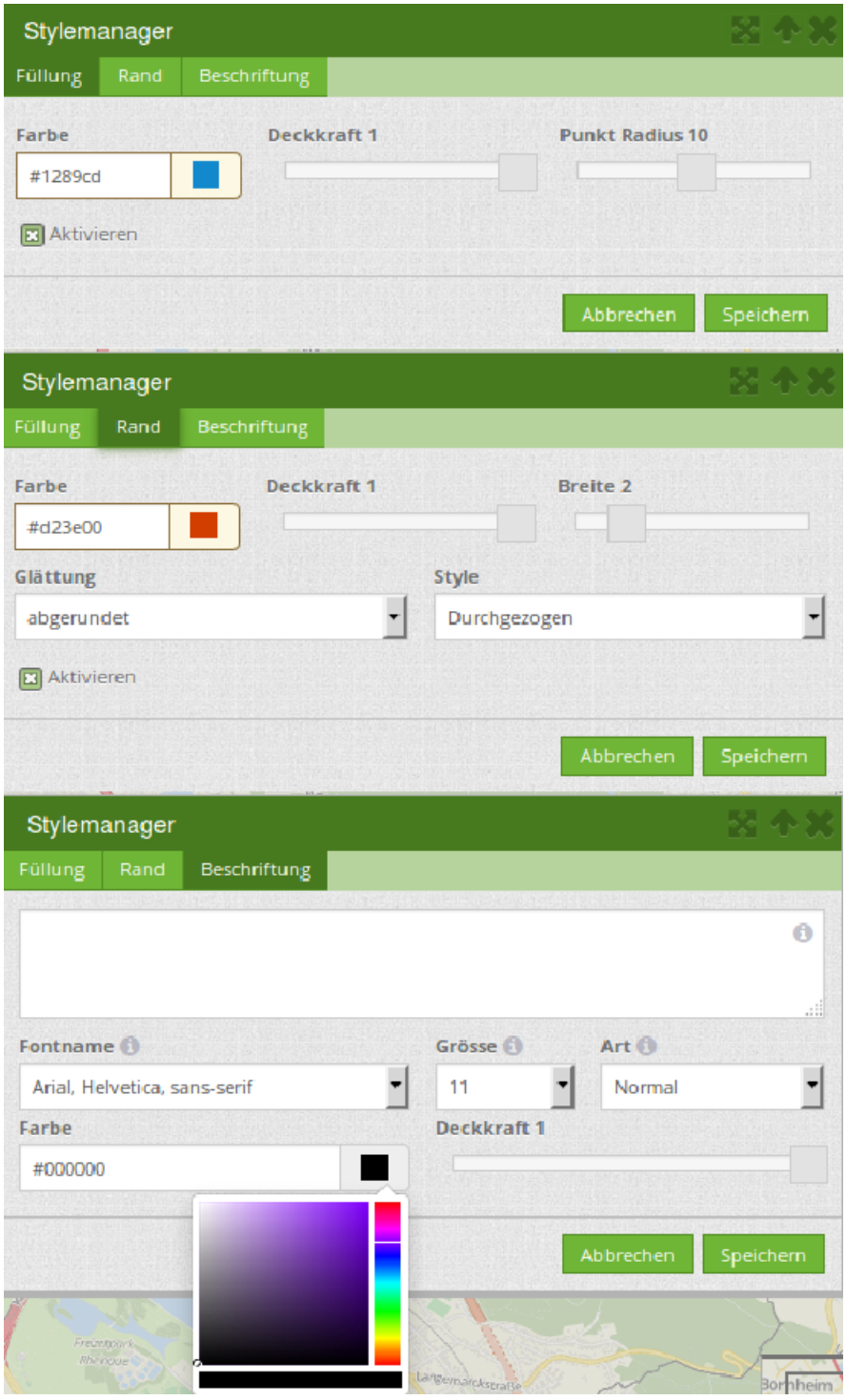
- **allowChangeVisibility:** Ändern der Sichtbarkeit von einem Treffer in der Karte (sichtbar/nicht sichtbar). [true/false]. Es wird ein Auge-Symbol zu jedem Feature eingeblendet, mit dem dieses explizit aus- und wieder eingeblendet werden kann.

- **showVisibilityNavigation:** Ändern der Sichtbarkeit von allen Treffern in der Karte (sichtbar/nicht sichtbar). [true/false]

Experimentell:

- **allowCustomerStyle:** Erlauben benutzerspezifischer Styles für Elemente in der Karte. [true/false]. Dieses Feature ist noch experimentell: Zu jedem Feature können über ein Symbol die einzelnen Styles verändert werden.

Es erscheint ein Style-Manager zur Anpassung einzelner Features.

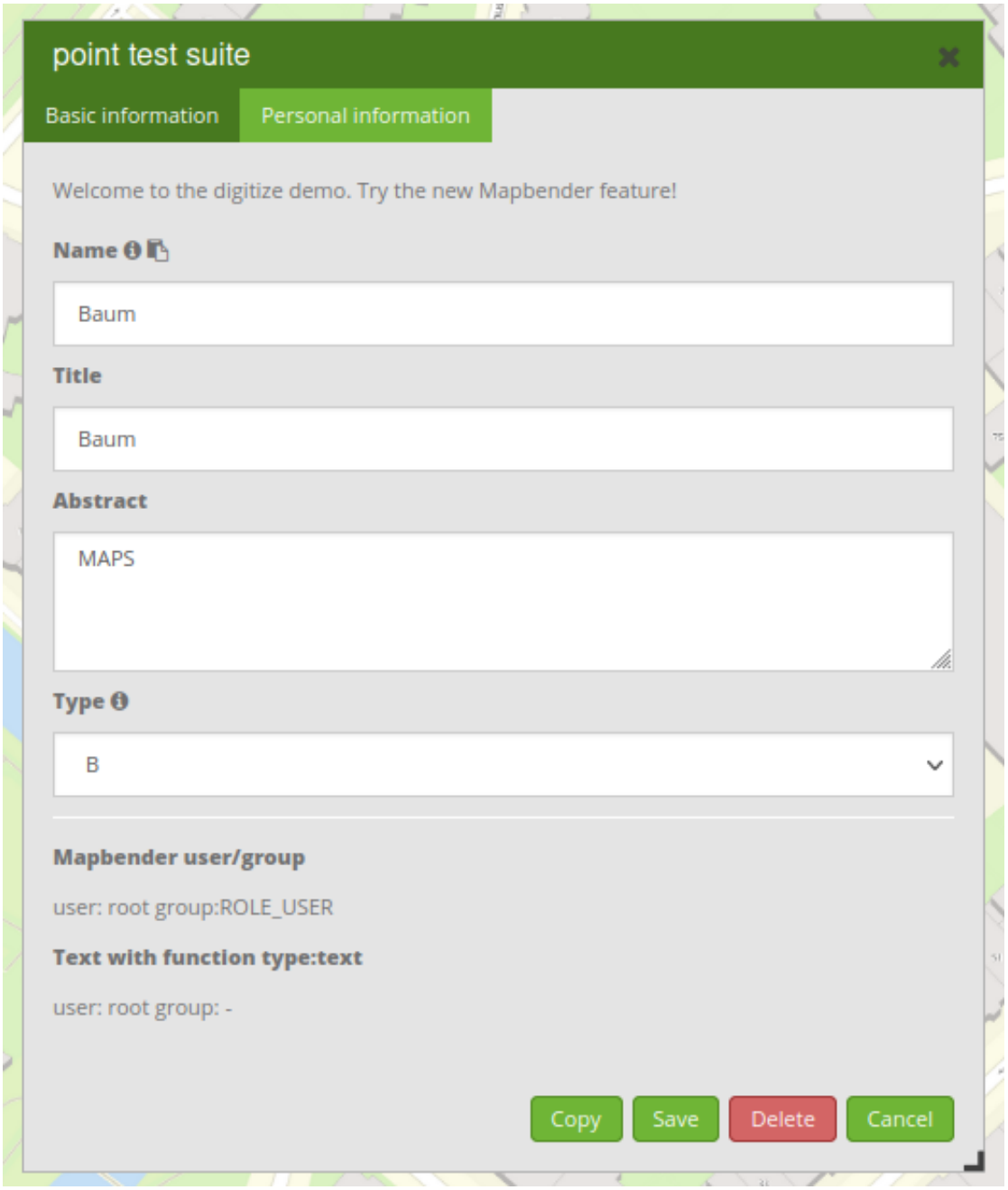

Stylemanager
Füllung
Rand
Beschriftung
Farbe
#1289cd
Deckkraft 1
Punkt Radius 10
Aktivieren
Abbrechen
Speichern
Stylemanager
Füllung
Rand
Beschriftung
Farbe
#d23e00
Deckkraft 1
Breite 2
Glättung
abgerundet
Style
Durchgezogen
Aktivieren
Abbrechen
Speichern
Stylemanager
Füllung
Rand
Beschriftung
Fontname
Arial, Helvetica, sans-serif
Grösse
11
Art
Normal
Farbe
#000000
Deckkraft 1
Abbrechen
Speichern

Definition Popup

In Zusammenhang mit der Digitalisierung können für die Erfassung von dazugehörigen Sachdaten komplexe Formulare generiert werden.

Folgende Optionen stehen für den Aufbau von Formularen zur Verfügung:

- Definition von mehreren Datenquellen und Geometrieformaten für die Erfassung. Die verschiedenen Quellen werden über eine Auswahlbox angeboten.
- Als Datenquelle wird eine Datenbank-Tabelle angesprochen. Es ist möglich, eine Auswahl der Daten über einen Filter heranzuziehen.
- Textfelder
- Textblöcke (mehrzeilige Textfelder)
- Selectboxen, Multiselectboxen (Füllen der Auswahlbox über eine feste Definition von Werten in der YAML-Definition oder über ein Select auf eine Tabelle)
- Checkboxen und Radiobuttons
- Datumsauswahl
- Dateiupload und Bildanzeige
- Definition von Reitern
- Definition von Trennlinien (breakLine)
- Definition von beschreibenden Texten zur Information
- Definition von Hilfetexten
- Pflichtfelder, Definition von regulären Ausdrücken für die Formatvorgabe bestimmter Feldinhalte
- Möglichkeit, in Formulare eingegebene Inhalte per Buttonklick in die Zwischenablage zu kopieren
- Karten-Refresh nach Speichern




```
popup:                                                  # Definition des Formularfensters als
↪PopUp-Fenster. Weitere experimentelle Anpassungsmöglichkeiten unter http://api.jqueryui.com/dialog/
    title: POI                                          # Definition des Titels vom
↪Formularfensters
    height: 400                                         # Höhe des Formularfensters
    width: 500                                          # Breite des Formularfensters

    #modal: true                                        # Alles außer dem Formularfensters wird
↪ausgegraut und die Position und Größe des Fensters ist für die Dauer der Datenaufnahme fixiert
    #position: {at: "left+20px",  my: "left top-460px"}  # Positionierung des Formularfensters
↪im Browserbereich
```

Definition der Objekttabelle

Der Digitizer stellt eine Objekttabelle bereit. Über diese kann auf die Objekte gezoomt werden und das Bearbeitungsformular kann geöffnet werden. Die Objekttabelle ist sortierbar. Die Breite der einzelnen Spalten kann optional in Prozent oder Pixeln angegeben werden.

- **tableFields: Definition der Spalten für die Objekttabelle.**
  - Definition einer Spalte: [Tabellenspalte]: {label: [Beschriftung], width: [css-Angabe, z.B. Angabe der Breite]}
- **searchType:** Suchbereich in der Karte, Anzeige aller Objekttreffer in der Tabelle oder nur aller Objekttreffer in dem derzeitigen Kartenausschnitt [all / currentExtent] (Standard: currentExtent).
- **showExtendSearchSwitch:** Anzeige der searchType Selectbox zur Suche im Kartenausschnitt aktivieren oder deaktivieren [true/false]
- **view: Einstellungen zu der Objekttabelle**
  - **type**: Templatename [table]
  - **settings**: Einstellungen zum Funktionsumfang der Objekttabelle *(neu hinzugefügt, noch nicht vollst. dokumentiert!)*

Detaillierte Informationen zu möglichen Angaben finden Sie unter https://datatables.net/reference/option/

```
searchType: currentExtent
showExtendSearchSwitch: true
tableFields:
    gid: {label: Nr. , width: 20%}
    name: {label: Name , width: 80%}
view:
    type: table
    settings:
        info: true
        processing: false
        ordering: true
        paging: true
        selectable: false
        autoWidth: false
        order: [[1, "asc"]]          # Spalte 1 | 2 vorsortieren
```

Dateireiter (type tabs)

Die Formularelemente können in verschiedenen Reitern dargestellt werden. Dazu dient das formItem type “tabs”.

```
formItems:
   - type: tabs                      # Type tabs erzeugt Reiter im Erfassungsformular
     children:                       # Die Reiter werden als Unterobjekte (children) vom Type form
↪definiert.
       - type: form
         title: Basic information    # Titel des Reiters
         css: {padding: 10px}
         children:                   # Durch mehrere Unterobjekte in Gruppen können Angaben im Formular
↪nebeneinander angeordnet werden.
             - type: label
```

(continues on next page)

(continued from previous page)

```
            title: Welcome to the digitize demo. Try the new Mapbender3 feature!
            ...
```

Bei jedem Eingabefeld können unabhängig vom Typ Verhaltensparameter per Event und Stylingangaben per CSS vergeben werden. Damit können beispielsweise wichtige Felder hervorgehoben oder auch ein Attributfeld in Abhängigkeit zu einem anderen Feld gefüllt werden.

Verhaltensparameter:

- load, focus, blur
- input, change, paste
- click, dblclick, contextmenu
- keydown, keypress, keyup
- dragstart, ondrag, dragover, drop
- mousedown, mouseenter, mouseleave, mousemove, mouseout, mouseover, mouseup
- touchstart, touchmove, touchend, touchcancel

```
formItems:
  - type: tabs
    children:
      - type: form
        [...]
            - type: input
              name: firstname
              title: Firstname
              css: {width: 30%}
              input: |
                   var inputField = el;
                   var form = inputField.closest(".modal-body");
                   var datenkennungField = form.find("[name='datenkennung']");
                   datenkennungField.val(inputField.val());
              focus: |
                   var inputField = el;
                   var form = inputField.closest(".modal-body");
                   form.css("background-color","#ffc0c0");
              blur: |
                   var inputField = el;
                   var form = inputField.closest(".modal-body");
                   form.css("background-color","transparent");
            - type: date
              name: date
              title: Datum
              css: {width: 30%}
              # Hervorhebung des Jahres bei Änderung des Datum-Feldes und autom. Füllen des Jahres aus
↪dem Datum
              change: |
                   var inputField = el;
                   var form = inputField.closest(".modal-body");
                   var yearField = form.find("[name='year']");
                   var year = inputField.val().match(/\d+$/)[0];
                   yearField.val(year);
                   yearField.css("background-color","#ffc0c0");
```

Textfelder (type input)

```
- type: input                                        # Typ Textfeld
  title: Title for the field                         # Beschriftung mit dem Titel des Feldes (optional)
  name: column_name                                  # Referenz zur Tabellenspalte
  copyClipboard: false                               # Button, der eingetragene Inhalte in die
↪Zwischenablage kopiert (optional) (Standard: false).
  mandatory: true                                    # Angabe ob Pflichtfeld (optional). [true/false]
  mandatoryText: You have to provide information.    # Text, sofern Pflichtfeld nicht gefüllt wurde
  infoText: 'Bitte geben Sie einen Wert an'          # Definition eines Informationstextes (optional)
  cssClass: 'input-css'                              # css-Klasse, die für das Textfeld verwendet
↪werden soll
  value: 'default Text'                              # Definition eines Standard-Wertes  (optional)
  placeholder: 'Bitte geben Sie ... ein'             # Platzhalter, der vor der Eingabe erscheint
↪(optional)
```

Auswahlboxen (selectbox oder multiselect [type select])

Durch die Definition einer Auswahlbox können vordefinierte Werte im Formular genutzt werden. Hier wird in eine Auswahlbox mit einem wählbaren Eintrag (type select) und einer Auswahlbox mit mehreren auswählbaren Einträgen (type multiselect) unterschieden.

(1) select - ein Eintrag kann ausgewählt werden

```
- type: select                                       # Typ Auswahlbox
  title: select some types                           # Beschriftung mit dem Titel des Feldes (optional)
  name: my_type                                      # Referenz zur Tabellenspalte
  copyClipboard: false                               # Button, der den ausgewählten Wert in die
↪Zwischenablage kopiert (optional).
  multiple: false                                    # Definition einer Mehrfachauswahl (multiselect).
↪(Standard: false)
  options:                                           # Definition der Optionen (key: value)
      1: pub
      2: bar
      3: pool
      4: garden
      5: playground
```

(2) multiselect - mehrere Einträge können ausgewählt werden

Multiselect-Box wird durch das attribute *multiple: true* aktiviert. Es können mehrere Einträge ausgewählt werden. Die Nutzung und die Voraussetzungen der Datenbank können variieren. Bei dem oben beschriebenen Beispiel können die Interessen in den POI als Multiobjekte abgespeichert werden. Das Datenbankfeld ist weiterhin ein character varying.

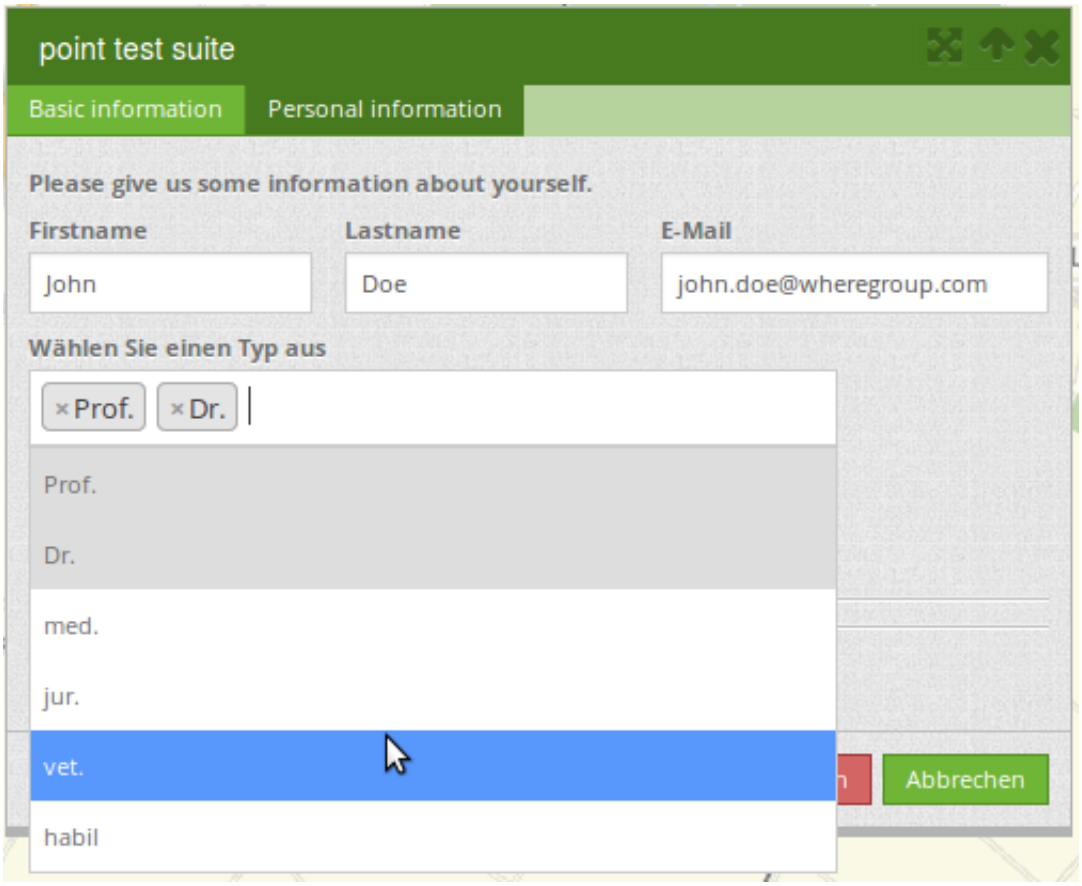
```
-
  type: select
  multiple: true
  title: Interests
  name: interests
  copyClipboard: false
  options:
    maps: maps
    reading: reading
    swimming: swimming
```

(continues on next page)

(continued from previous page)

```
    dancing: dancing
    beer: beer
    flowers: flowers
```

Anmerkungen: Ab Digitizer 1.2 wird für die Mehrfachauswahl eine vereinfachte Auswahlmöglichkeit genutzt, die auch die Suche innerhalb der Drop-Down-Liste ermöglicht. Die Navigation durch die Liste ist mit der Tastatur möglich. Mögliche Einträge werden beim Eintippen hevorgehoben. Eine vorhandene Auswahl kann über das kleine "x" Symbol wieder deaktiviert werden. Aktuelle Informationen zur Digitizer-Version finden sich unter: https://github.com/mapbender/mapbender-digitizer



Das SQL (wenn maps und reading abgespeichert wurden):

```
gisdb=> select interests from poi where gid=3;
interests
--------------
maps,reading
(1 row)
```

Beim Abspeichern von Einträgen werden die Schlüsselwörter in der Datenbank abgelegt (Bsp.: "dancing: Tanzen" und "flowers: Blumen" speichert "dancing,flowers"). Das Kopieren mehrerer Werte in die Zwischenablage ist ebenfalls möglich (siehe konfiguriertes Beispiel).

```
- type: select                      # Typ Auswahlbox
  title: Wählen Sie einen Typ aus   # Beschriftung mit dem Titel des Feldes (optional)
  name: my_type                     # Referenz zur Tabellenspalte
```

(continues on next page)

(continued from previous page)

```
  copyClipboard: true                # Button, der den ausgewählten Wert in die Zwischenablage kopiert
↪(optional).
  multiple: true                     # Definition einer Mehrfachauswahl (multiselect). (Standard: false)
  options:                           # Definition der Optionen (key: value)
    a: a
    b: b
    c: c
```

Füllen der Auswahlboxen über eine SQL Abfrage

Mit einer SQL-Abfrage können die Werte direkt aus der Datenbank gezogen werden. Dabei ist zu beachten, dass die key-value-Zuordnungen wegfallen und nur noch die Stützpunkte abgespeichert werden.

```
- type: select                       # Typ Auswahlbox
  title: select some types           # Beschriftung (optional)
  name: my_type                      # Referenz zu Tabellenspalte
  connection: connectionName         # Definition einer Datenbankverbindung (connection)
  sql: 'SELECT DISTINCT key, value FROM tableName order by value' # Definition SQL, Abfrage der Werte
↪key und value
```

Texte/Label (type label)

```
- type: label                                  # Typ Label schreibt einen nicht bearbeitbaren Text
↪in das Formularfenster.
  title: 'Please give information about the poi.' # Definition eines nicht bearbeitbaren Textes.
```

Texte (type text)

Im Formular können Texte definiert werden. Hierbei kann auf Felder der Datenquelle zugegriffen werden, dazu wird JavaScript verwendet.

```
- type: text                    # Typ text zur Generierung von dynamischen Texten aus der Datenbank
  title:       Name             # Beschriftung (optional)
  name:        name             # Referenz zu Tabellenspalte, dessen Inhalt angezeigt werden soll
  css:         {width: 80%}     # CSS Definition (optional)
  text: data.gid + ': ' + data.name
  # Text Definition in JavaScript
  # data - Angabe, dass Datenbankfeld aus der Tabelle angesprochen wird.
  # z.B.: data.gid --> Anzeige der ID der Geometrie im Textfeld
```

Textbereiche (type textArea)

Ähnlich zum Textfeld über type input (siehe oben) können hier Textbereiche erzeugt werden, die bei type textArea mehrere Zeilen umfassen können.

```
- type: textArea                   # Typ textArea erzeugt einen Textbereich
  rows: 4                          # Anzahl der Zeilen für den Textbereich, die beim Öffnen des
↪Formulars erscheinen. Feld kann per Maus im Formular größer gezogen werden.
  name: beschreibung               # Tabellenspalte
  title: Bestandsaufnahme Bemerkung # Beschriftung (optional)
```

Trennlinien (type breakLine)

```
- type: breakLine      # fügt eine einfache Trennlinie ein
```

Checkboxen (type checkbox)

```
- type:  checkbox        # Typ checkbox erzeugt eine Checkbox. Beim Aktivieren wird in die Datenbank
↪der angegebene Value (hier 'TRUE') geschrieben.
  title: Is this true?   # Beschriftung (optional)
  name:  public          # Referenz zu Tabellenspalte
  value: true            # angegebener Parameter beim Aktivieren der Checkbox wird in DB gespeichert
↪(hier 'TRUE').
  checked: false         # Definiert, ob die Checkbox vordefiniert angehakt sein soll. (Standard: false)
```

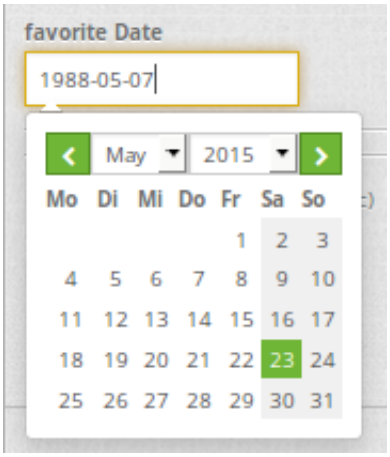
Pflichtfelder

Die Hinweise für ein Pflichtfeld erscheinen über dem jeweiligen Feldern. Bei einer fehlenden Angabe in einem definierten Pflichtfeld wird dieses rot umrandet und (wenn vorher definiert) erscheinen Hinweise. Das Objekt kann nicht gespeichert werden, wenn Pflichtangaben fehlen.

Note: Hinweis: Bei der Nutzung von mehreren Reitern in dem Formular kann es sein, dass der Erfasser bei einem Pflichtfeld auf einem nicht sichtbaren Reiter eine Angabe falsch setzt und das Abspeichern daher nicht funktioniert. Hier erscheint keine Fehlermeldung außerhalb des Formulars. Der Erfasser muss die Angaben in dem Formular überprüfen (Kennzeichen: rote Umrandung/Sprechblase mit Hinweis), bevor diese korrekt abgespeichert werden können.

```
- type:  [Angabe zum Feldtyp]             # jedes Feld kann zum Pflichtfeld gemacht werden

  mandatory: true                         # true - Das Feld muss gefüllt werden. Ansonsten kann der
↪Datensatz nicht gespeichert werden. Bei der Definition sind auch reguläre Ausdrücke möglich.
  mandatorytitle: Pflichtfeld füllen!     # Text der im Feld angezeigt wird, wenn das Feld nicht
↪gefüllt wird oder mit einem ungültigen Wert gefüllt wird.
  mandatoryText: Bitte eine Zahl eingeben! # Text der in einer Sprechblase über dem Feld angezeigt wird,
↪ wenn das Feld beim Speichern nicht gefüllt ist oder mit einem ungültigen Wert gefüllt ist.
  mandatory: /^\w+$/gi                    # Es können auch reguläre Ausdrücke angegeben werden, um die
↪Eingabe zu überprüfen (z.B. Email oder numbers) Weitere Informationen unter: http://wiki.selfhtml.org/
↪wiki/JavaScript/Objekte/RegExp

  # Prüfung, ob die Eingabe eine Zahl ist
  mandatory: /^[0-9]+$/
  mandatoryText: Bitte eine Zahl eingeben!
```

Datumsauswahl (type date)




```
- type: date                                    # Textfeld, das eine Datumsauswahl bereitstellt
  title: favorite Date                          # Beschriftung (optional)
  name: date_favorite                           # Referenz zu Tabellenspalt
  mandatory: true                               # Pflichtfeld
```

Hilfetexten zu den Eingabefeldern (Attribut infoText)

Anders als bei Hifetexten zu den Pflichtfeldern kann der Infotext über jedem Feld erscheinen, unabhängig davon, ob dieses ein Pflichtfeld ist oder nicht. Bei der Angabe infotext: [Text] erscheint ein Info-Button über dem jeweiligen Feld. Der Klick auf diesen Button öffnet den angegebenen Informationstext.

```
- type:  [Angabe zum Feldtyp]                                # jedes Feld kann einen Infotext nutzen

  infoText: In dieses Feld dürfen nur Zahlen eingegeben werden  # Hinweistext, der angezeigt wird über
↪i-Symbol.
```

Gruppierungen (type: fieldSet)

Elemente können in einer Zeile gruppiert werden, um logische Einheiten zu bilden oder um Platz zu sparen. Hierbei muss ein fieldSet definiert werden. Anschließend können die Elemente der Gruppe unter children angegeben werden. Für jedes Gruppenelement kann eine Breite über CSS angegeben werden, um die Aufteilung der Zeile für die angegebenen Elemente zu kontrollieren.

```
- type: fieldSet            # Gruppierung von Feldern, unabhängig vom Feldtyp
  children:                 # Angabe der Gruppenelemente unter children
      - type: input
        title: Vorname
        name: firstname
        css: {width: 30%}   # Angabe der Breite des Gruppenelements. Zusammen sollten die Elemente 100%
↪ergeben.
      - type: input
        title: Nachname
        name: lastname
        css: {width: 30%}
```

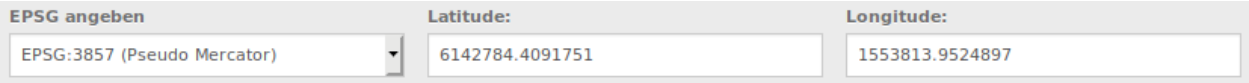
(continues on next page)

(continued from previous page)

```
    - type: input
      title: E-Mail
      name: email
      css: {width: 40%}
```

Koordinaten (type coordinates)

Um ein visuelles Feedback zu erhalten, Koordinaten manuell eingeben zu können oder diese manuell nachzubessern, kann für Punkt-Geometrien das *coordinates* Element eingebunden werden.




```
- type: coordinates
  title_epsg: 'EPSG angeben'
  title_longitude: 'Longitude'
  title_latitude: 'Latitude'
  coordinatesFieldsOrder: ['epsg','x','y']
  title: ''
  css: {width: 100% }
  epsgCodes: [['EPSG:3857', 'EPSG:3857 (Pseudo Mercator)'], ['EPSG:4326', 'EPSG:4326 (WGS84)']]
```

Die verfügbaren Parameter sind:

- **title_epsg**: die Beschriftung für das Feld zur Auswahl des Koordinatensystems der Koordinaten
- **title_longitude**: die Beschriftung für das logitude Feld
- **title_latitude**: die Beschriftung für das latitude Feld
- **coordinatesFieldsOrder**: eine Liste aus Strings, die die Reihenfolge der drei Eingabefelder (Koordinatensystemauswahl, Longitude, Latitude) definiert
- **epsgCodes**: eine Liste aus Tupeln, die jeweils einen EPSG Code und dessen Beschriftung im Formular definieren

Dateiupload (type file)

Über den Dateiupload können Dateien durch die Angabe in einer Datenbankspalte im Formular verknüpft werden. Dazu werden die hochgeladenen Dateien im Mapbender gespeichert und der Pfad in der Spalte vermerkt.

Der Speicherpfad und der Name der abgespeicherten Dateien kann bis jetzt nicht verändert werden. Der Dateiupload speichert immer in das gleiche Verzeichnis und baut sich aus den Parametern:

- Tabellenname
- Spaltenname
- Dateiname

auf.

Das Verzeichnis ist:

- `<mapbender>/web/uploads/featureTypes/[tabellenname]/[spaltenname]/[dateiname].png`

Die in der Datenbank verlinkte URL ist:

- `http://localhost/mapbender/uploads/featureTypes/[tabellenname]/[spaltenname]/[dateiname].png`

```
- type: file                        # Typ file für das Hochladen von Dateien
  title: Dateiupload                # Beschriftung (optional)
  text: Laden Sie ein Bild hoch.    # Informationstext zum Feld (optional)
  name: file_reference              # Angabe der Datenbankspalte, in die der Speicher-Pfad geschrieben
↪wird


  # Experimentelle Parameter:
  #accept: image/*                  # Vorauswahl von Elementen im Image-Format (Fenster für Dateiupload
↪öffnet sich mit Einschränkungsfilter)
                                    # Es können jedoch weiterhin auch andere Dateiformate hochgeladen
↪werden.
```

Anmerkungen: Es wird momentan noch ein “thumbnail” Verzeichnis angelegt, dass eine kleinere Version der Bilder beinhaltet. Dieses wird in weiteren Entwicklungsphasen noch verändert.

Für die Ansicht von hochgeladenen Bildern kann die Bildanzeige dazugenommen werden.

Bildanzeige (type image)

Für die Ansicht eines Bildes im Formular kann das Bild-Element genutzt werden. Durch die Angabe einer URL in einem Datenbankfeld oder einer URL über den src-Parameter können Bilder angezeigt werden.

Bilder, die durch das Element Dateiupload in einer Tabellenspalte vermerkt sind können somit auch direkt eingebunden und angezeigt werden.

Das Bild lässt sich durch die Angabe von den beiden Parametern src und name angeben.

- **src**: Url-Pfad oder Dateipfad (kann relativer Pfad sein)
- **name**: Url-Pfad oder Dateipfad wird aus der Tabellenspalte übernommen (kann kein relativer Pfad sein)
- Angabe von **name** und **src** zusammen: Der Inhalt der Datenbankspalte aus name wird genommen. Falls die Spalte leer ist wird die src-Angabe genutzt.

```
- type: image                                    # Type image für das Anzeigen von Bildern
  name: file_reference                           # Referenz zur Datenbankspalte. Wenn definiert,
↪wird der Pfad oder die URL in dem Feld ermittelt und "src" Option ersetzt
  src: "../bundles/mapbendercore/image/logo_mb3.png"  # Angabe eines Pfades oder URL zu einem Bild.
↪Falls der relative Pfad genutzt wird, muss relative: true stehen.
```

(continues on next page)

(continued from previous page)

```
  enlargeImage: true                                # Bild wird beim Klick auf das Vorschaubild auf
↪Originalgröße/maximale Auflösung vergrößert. Es wird nicht auf die Bildschirmgröße skaliert.

  # Experimentelle Angaben zum Styling
  imageCss:
    width: 100%                                     # Image CSS Style: Skaliert das Vorschaubild in
↪dem Formular, abweichend von der Originalgröße in Prozent.
```

Achtung: Wenn nur name und nicht name und src angegeben wird, erscheint bei leeren Spalteneinträgen ein Bild aus dem vorherigen Dateneintrag.

Dynamische Pfade (z.B. "bundles/mapbendercore/image/[nr].png" oder 'bundles/mapbendercore/image/' + data.image_reference) können nicht angegeben werden.

Eine Möglichkeit, das zu umgehen, wäre ein Trigger, der in die Datenbankspalte beim Insert den Pfad und den Inhalt eines Tabellenfeldes als Name zusammenführt.

Definition der zur Verfügung stehenden Werkzeuge (Toolset Type)

Jeder FeatureType kann unterschiedliche Toolsets freischalten, die dann in der Schaltflächenleiste des Digitizers sichtbar sind. In dem FeatureType "poi" wird mit dem "drawPoint" Toolset z.B. die Schaltfläche zum Erstellen eines neuen Punktes freigeschaltet, mit dem Toolset "modifyFeature" die Verschieben-Schaltfläche eingeblendet.

Werkzeugliste:

- **drawPoint** - Punkt erstellen
- **drawLine** - Line erstellen
- **drawPolygon** - Polygone erstellen
- **drawRectangle** - Rechteck erstellen
- **drawCircle** - Circle erstellen
- **drawEllipse** - Ellipse erstellen
- **drawDonut** - Donut erstellen oder die bestehende Geometrien editieren
- **modifyFeature** - einzelne Knotenpunkte bei Geometrien verschieben
- **moveFeature** - Geometrien verschieben
- **selectFeature** - Geometrien de-/selektieren (experimentell). Es findet noch keine Interaktion mit der Tabelle statt und die zur Verfügung stehenden Workflows beschränken sich auf das Löschen.
- **removeSelected** - die selektierten Geometrien löschen (experimentell). Löscht alle in der Karte selektierten Objekte.
- **removeAll** - Vorsicht: alle Geometrien aus der Tabelle löschen

Definition der für die Erfassung verwendeten Toolset Typen:

```
polygon:
    [...]
    toolset:
        - type: drawPolygon
        - type: drawRectangle
```

(continues on next page)

(continued from previous page)

```
        - type: drawDonut
        - type: removeSelected
```

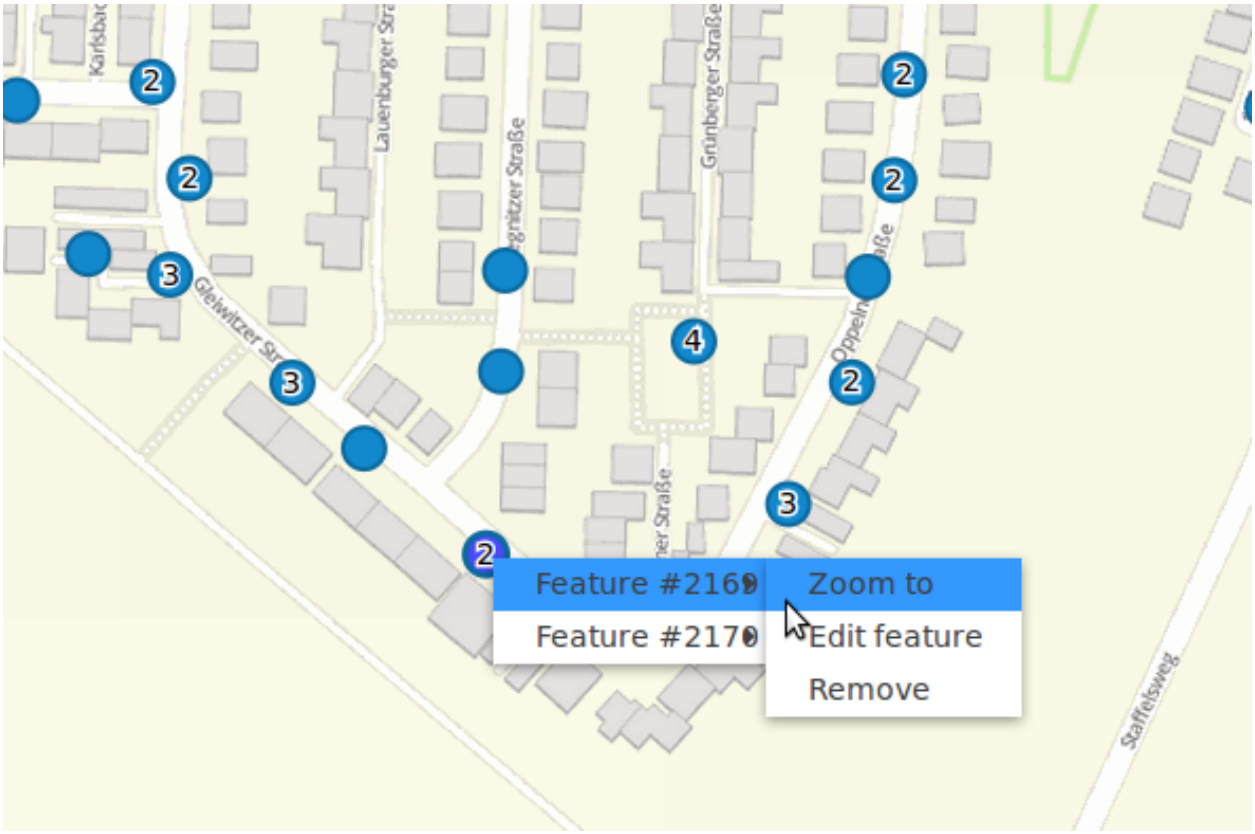
Suche in den Tabellen (inline Search)

Über die Suche können Begriffe in der Tabelle gesucht werden. Die aktivierte Sucheleiste erscheint über der Tabelle und nach der Eingabe eines Suchbegriffs werden alle Spalten dieser Tabelle durchsucht und deren Ergebnisse angezeigt.

```
poi:
    ...
    inlineSearch: true      # Suche in den Tabellenspalten (Standard: true)
    ...
```

Kontextmenü

Über das Kontextmenü kann ein Objekt auf der Karte näher betrachtet werden. Nach der Aktivierung lässt sich über den rechten Mausklick auf ein Objekt ein Kontextmenü öffnen.



Elemente des Kontextmenüs:

- **Heranzoomen:** Auf den Kartenausschnitt des Objekts zoomen.
- **Bearbeiten:** Informationen zu dem Objekt verändern. Klick öffnet die Digitalisieroberfläche.

- **Löschen:** Löscht das ausgewählte Objekt.

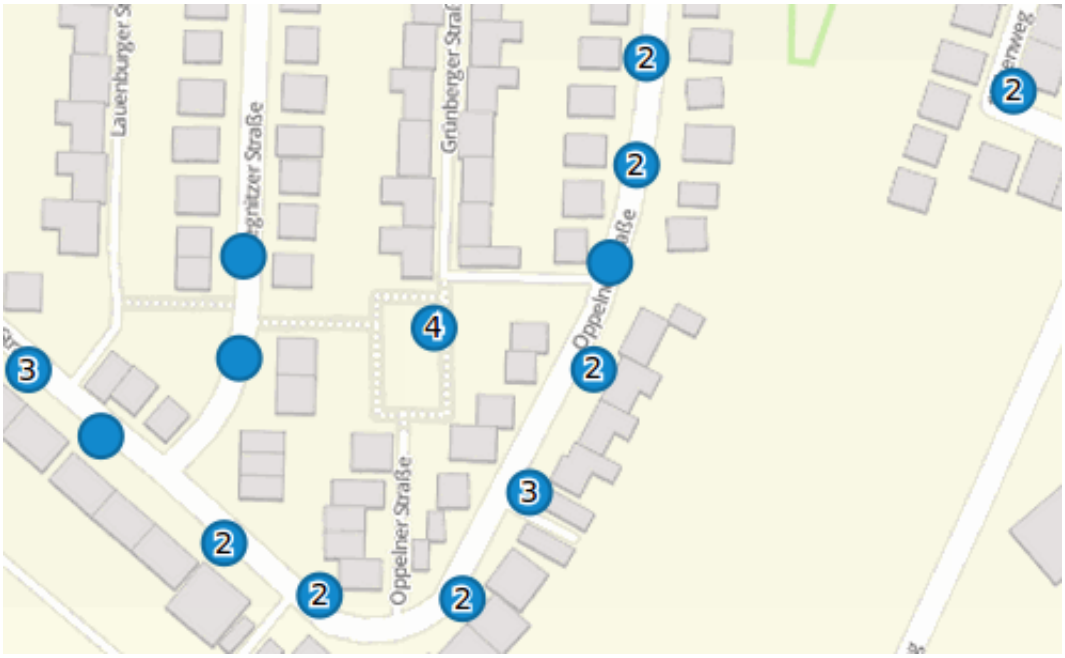
Ist die korrespondierende *Basisdefinition* (allowEdit, allowDelete) nicht definiert, stehen diese auch im Kontextmenü nicht zur Verfügung. In dem oben stehenden Beispiel steht für die Polygone die Löschen-Funktion nicht zur Verfügung.

```
poi:
    ...
    useContextMenu: true
    ...
```

Clustering (experimentell)

Über das Clustering können die Objekte auf der Karte zusammengefasst werden. Abhängig von der definierten Distanz und Zoomstufe werden unterschiedlich viele Objekte zusammengeführt.

Durch die Komplexität des Clusterings können sich in zukünftigen Versionen noch Änderungen an der Funktion und der Syntax ergeben, so dass wir dieses Feature bislang als experimentell kennzeichnen. Abhängigkeiten existieren zu der Anzeige der Features für den aktuellen Ausschnitt/den gesamten Bereich und den unterschiedlichen Geometrietypen.



Definition der Clusterelemente:

- **scale:** Zoomstufe.
- **distance:** Distanz zwischen einzelne Features in Metern, die zusammengefügt werden.
- **disable:** Schaltet Clustering für die Zoomstufe ab.

```
poi:
    [...]
    clustering:
        -
            scale: 10000        # Zoomstufe
            distance: 60        # Distanz zwischen einzelne Features in Metern, die zusammengefügt
↪werden
```

(continues on next page)

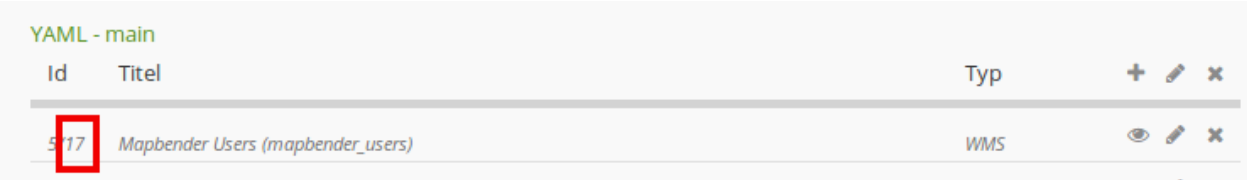
(continued from previous page)

```
        -
            scale: 2500
            distance: 40
        -
            scale: 1000
            distance: 20
        -
            scale: 500
            distance: 1
            disable: true       # Schaltet Clustering für die Zoomstufe ab
    [...]
```

Karten-Refresh nach Speichern

Nach dem Speichern eines Objekts kann ein Refresh der Karte über die Option `refreshLayersAfterFeatureSave` aktiviert werden. Über diesen Parameter werden die definierten Layer-Instanzen aus dem Map-Element neu geladen. Damit werden Änderungen in WMS-Diensten direkt in der Karte sichtbar. Dieses Event wird nur gestartet, wenn der Speichern Knopf aus dem Attributdialog verwendet wird.

Wenn YAML-Anwendungen unter /application genutzt werden, kann die Angabe per eindeutigen Name oder per Instance-ID erfolgen. Wenn die Anwendungen über die graphische Oberfläche im *Backend* mit dem Digitizer-Element bearbeitet wird, kann nur die Angabe per Instance-ID erfolgen.




```
poi:
    [...]
     allowEdit: true
     refreshLayersAfterFeatureSave:  # bei keiner Angabe in diesem Bereich erfolgt kein Karten-Refresh
↪nach Speichern
       - 17
       - 18
       - osm                         # Namensangabe nur bei Anwendungen unter applications/ möglich
    [...]
```

Anmerkungen: Karten-Refresh nach dem Speichern ist ab Version 1.2 möglich. Mehr Informationen unter: https://github.com/mapbender/mapbender-digitizer

Duplizieren von Objekten

Bereits erfasste Objekt können dupliziert werden. Dies geht über einen Duplizieren-Button innerhalb des Erfassungsfensters des aktuellen selektierten bereits vorhandenen Features, über das Context-Menu und die Treffertabelle. Damit das neue Objekt in der Karte besser erkannt werden kann ist hier eine farbliche Hervorhebung definierbar.

Der Duplizieren-Button kann in Abhängigkeit von einem bestimmten Attribut-Wert aktiviert werden. Dies bedeutet, dass nur wenn das entsprechende Attribut einen bestimmten Wert hat (date > 0), erscheint die Duplizieren-Funktion.

- **data**: Angabe von Standardwerten für Attributfelder
- **rules**: regelbasiertes Duplizieren (nur wenn die Regel zutrifft darf das Objekt dupliziert werden).
- **style**: Styling des kopierten Objekts (mehr dazu s.u. unter dem Bereich Darstellung)
- **on**: Events beim Duplizieren

```
poi:
    [...]
     copy: # bei keiner Angabe in diesem Bereich können keine Objekte dupliziert werden
       enable: true
       data:
         date: 2017
       rules:
         - feature.attributes.id > 10
       style:
         label: "Dupliziertes Objekt"
         fillColor: "#ff0000"
         fillOpacity: 1
         strokeWidth: 4
         strokeColor: "#660033"
       on:
         success: widget._openFeatureEditDialog(feature)
         error: console.error(feature)
```

Events

Es gibt mehrere Events, die zu einem Feature zugeordnet werden können, um Attribute vor oder nach der Aktion zu manipulieren.

- **onBeforeSave**: Event vor dem Speichern von neuen/veränderten Informationen
- **onAfterSave**: Event nach dem Speichern von neuen/veränderten Informationen
- **onBeforeUpdate**: Event vor der Aktualisierung von veränderten Informationen
- **onAfterUpdate**: Event nach der Aktualisierung von veränderten Informationen
- **onBeforeSearch**: Event vor dem Suchen in SearchField des Digitizers
- **onAfterSearch**: Event nach dem Suchen in SearchField des Digitizers
- **onBeforeRemove**: Event vor dem Löschen von Daten
- **onAfterRemove**: Event nach dem Löschen von Daten

Im Unterschied zu den Save-Events arbeiten die Update-Events nur bei einer Aktualisierung der Daten, nicht bei einer Erstellung.

Anmerkung: Die Events sind noch in der Entwicklung und sollten mit Voraussicht eingebunden werden. Die korrekte Abstimmung der Events aufeinander und ihre Abhängigkeiten sind noch nicht vollständig fertiggestellt und können sich in zukünftigen Versionen ändern.

Im folgenden einige Anwendungsbeispiele. Falls mehrere Parameter in einem Event gesetzt werden sollen können diese durch ein Simikolon getrennt hintereinander aufgelistet werden, z.B.

```
events:
  onBeforeSave: $feature->setAttribute('interests', 'maps'); $feature->setAttribute('name', 'test');
```

Speichern von festen Sachdaten in zusätzlichen Attributspalte:

Das folgende Beispiel zeigt, wie Daten beim Speichern in eine zusätzliche Attributspalte geschrieben werden können. Hier geschieht das mit der Spalte “interests” und dem festen Wert “maps”. Beim Speichern wird der feste Wert in die Tabelle gespeichert und kann z.B. über einen Filter für die selektierte Anzeige genutzt werden.

```
events:
  onBeforeSave: $feature->setAttribute('interests', 'maps');
```

Speichern von Gruppenrollen in zusätzlichen Attributspalte:

Das folgende Beispiel zeigt, wie Mapbender-Benutzerdaten beim Speichern in eine zusätzliche Attributspalte geschrieben werden können. Hier geschieht das mit der Spalte “group” und der Füllung mit den jeweiligen Gruppenrollen des Benutzers (userRoles)

```
events:
  onBeforeSave: $feature->setAttribute('group', implode(',', $userRoles));
```

Speichern von Sachdaten in zusätzlichen Attributspalten:

Das folgende Beispiel zeigt, wie Daten beim Speichern in eine zusätzliche Attributspalte geschrieben werden können. Hier geschieht das mit den Spalten “geom” und “geom2”. Beim Speichern sollen die Daten von geom in das Feld geom2 geschrieben werden.

Man kann das Event je nach Anwendungsfall bei onBeforeInsert oder onBeforeUpdate eintragen.

Da zum Zeitpunkt des Editierens die Geometrie noch nicht persistent in der Datenbank ist, kann auf sie nicht als Feature zugegriffen werden, sondern nur über das jeweilige “Item”, eine interne Digitizer Speicherstruktur. Diese “Items” orientieren sich am Formular und den dort angegebenen Attributen.

```
events:
  onBeforeInsert: $item['geom2'] = $item['geom'];
  onBeforeUpdate: $item['geom2'] = $item['geom'];
```

Bei dem Event wird der Wert des Feldes “geom2” mit dem Wert des Feldes “geom” überschrieben.

Speichern unterschiedlicher Geometrietypen:

Dieses Szenario ist zu einem konsturierten Beispiel erweiterbar, in dem gleichzeitig unterschiedliche Geometrietypen geschrieben werden. Mithilfe von PostGIS können Linien in Punkte interpoliert werden. Im Digitizer kann ein Event genutzt werden, um das richtige SQL Statement abzuschicken.

```
events:
  onBeforeInsert: |
    $sql = "SELECT
    ST_Line_Interpolate_Point('".$item['geomline']."'::geometry, 1) as geom";
    $stmnt = $this->getConnection()->prepare($sql);
    $stmnt->execute();
    $result  = $stmnt->fetchAll();
    $item['geompoi'] = $result[0]['geom'];
```

Hier wird das onBeforeInsert-Event genommen. Der Längsstrich ‘|’ hinter dem Event zeigt einen mehrzeiligen Block an. Dieser Block besteht aus PHP-Code, der ein SQL-Statement weiterleitet. Das SQL Statement ruft die ST_Line_Interpolate_Point Funktion auf und übergibt die Geometrie der gezeichneten Linie. Da diese noch nicht persistent ist, muss über das “Item” auf die Geometrie (geomline) zugegriffen werden. Die restlichen Zeilen bauen das SQL Statement zusammen und schicken es an die im FeatureType angegebene SQL-Connection. In der letzten Zeile wird der resultierende Punkt (geompoi) in die Punktgeometrie geschrieben.

Buttons

Für die Erfassungsformulare können weitere Buttons definiert werden. Die Events beim Klick auf den Button können frei per JavaScript definiert werden. Somit können beispielsweise mailto-Angaben für die Einbindung einer Mail generiert werden.

```
poi:
    ...
      popup:
          title: polygon test suite
          buttons:
            - text: Nachricht an Bearbeiter
              click: |
                var body = encodeURI("Sehr geehrter Herr/Frau xx,"+"\nLink:"+location.href);
                location.href = "mailto:vorname.nachname@mail.com?subject=Neue Bearbeitung im WebGIS&
↪body=Mail an den Bearbeiter für die die weitere Bearbeitung.";
            - text: Nachricht an Kontrolle
              click: |
               location.href = "mailto:vorname.nachname@mail.com&subject=WebGIS&body=really?";
```

Darstellung (Styles)

Über die Angabe eines Styles kann definiert werden, wie die Objekte angezeigt werden.

- **default**: Definiert die normale Darstellung der Objekte auf der Karte
- **select**: Darstellung der ausgewählten Objekte beim Mouseover
- **selected**: Darstellung der ausgewählten Objekte nach dem Klick Event

```
poi:
    ...
    styles:
        default:
            graphic: true
            strokeWidth: 5
            strokeColor: "transparent"
            fillColor:  '#c0c0c0'
            fillOpacity: 1
            fillWidth: 2
            # label: ${name} ${type}
            # labelOutlineColor: '#eeeeee'
            pointRadius: 10
        select:
            strokeWidth: 1
            strokeColor: "#0e6a9e"
            fillOpacity: 0.7
            fillColor: "#0e6a9e"
            label: ${name} ${type}
            pointRadius: 10
        selected:
            strokeWidth: 4
            strokeColor: "#648296"
            fillOpacity: 1
            fillColor: "#eeeeee"
            label: ${name} ${type}
```

(continues on next page)

(continued from previous page)

```
            pointRadius: 10
    ...
```

- **strokeColor:** Farbe der Umrandungslinie [Farbwert/transparent]
- **strokeWidth:** Breite der Umrandungslinie [numeric]
- **strokeOpacity:** Transparenz der Umrandungslinie [0-1]
- **fillColor:** Farbe der Füllung [Farbwert/transparent]
- **fillWidth:** Breite der Füllung [numeric]
- **fillOpacity:** Transparenz der Füllung [0-1]
- **pointRadius:** Radius um den Mittelpunkt [numeric]
- **label:** Beschriftung des Objekts mit festen Werten und/oder DB-Feldern, z.B. "ID ${nummmer}"
- **labelOutlineColor:** Farbe der Umrandung von der Beschriftung [Farbwert/transparent]

YAML-Definition für das Element "digitizer" in der Sidepane in der mapbender.yaml

Diese Vorlage kann genutzt werden, um das Element in einer YAML-Anwendung einzubinden.

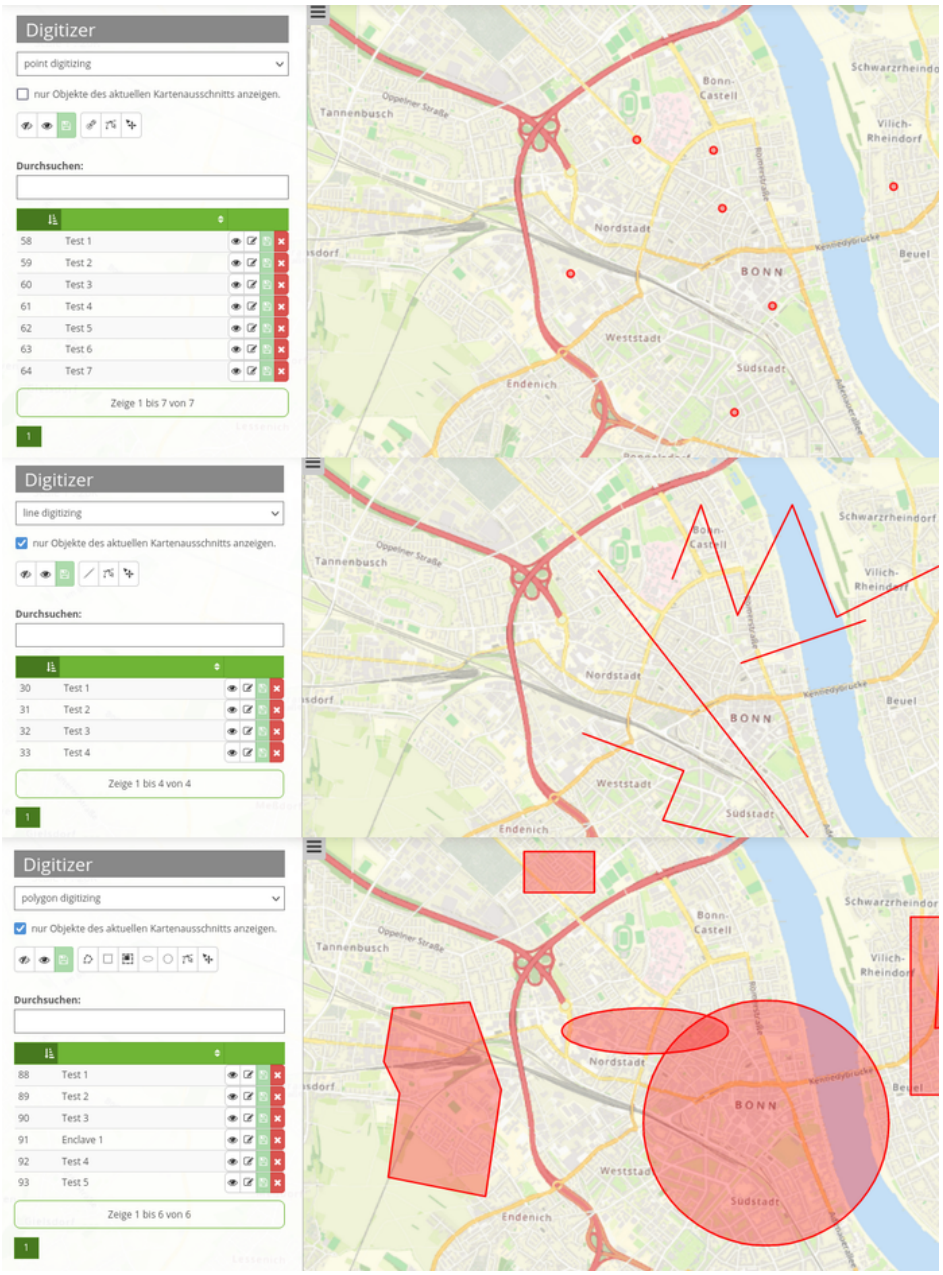
```
sidepane:
    digitizer:
        class: Mapbender\DigitizerBundle\Element\Digitizer
        title: Digitalisation
        target: map
        schemes:
            ...
```

Mehr Informationen zur Nutzung des Digitizers sind hier zu finden:

Funktionen des Digitizers

Der Digitizer ermöglicht das Editieren von FeatureTypes. Diese basieren auf Punkt-, Linien- oder Polygongeometrien und ihren Sachdaten. Letztere werden in dem Formular des Digitizers angezeigt. Das Editieren der Geometrien geschieht über die Karte. Der Digitizer ermöglicht eine Vielzahl von Funktionen zur Erstellung und Bearbeitung von Geometrien:

- Erstellen von Punkten, Linien und Polygonen (Quadrate, Rechtecke, Kreise und Ellipsen)
- Verschieben von Geometrieobjekten
- Einfügen von Stützpunkten (bei Linien und Polygonen)
- Erfassung von Polygonen mit Enklaven



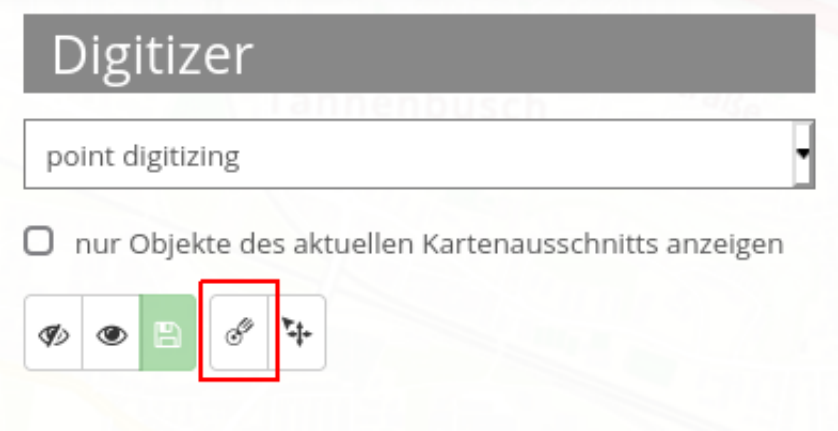
Die folgenden Abschnitte stellen die Arbeit mit Digitizer gemäß der Standardkonfiguration genauer vor.

Geometrien erstellen

In der Standardkonfiguration kann der Nutzer über ein Dropdown-Menü zwischen drei verschiedenen Geometrietypen wählen: Punkt, Linie und Polygon.

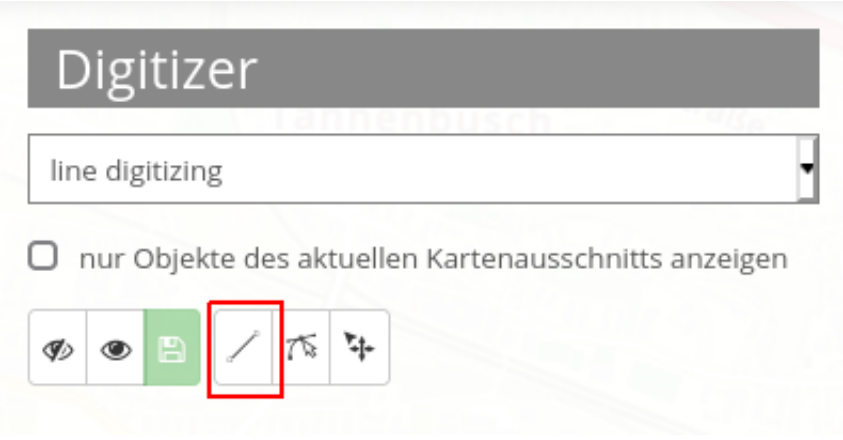
Punkte

Durch einen Klick auf den Button *"Punkt erstellen"* wird die Funktion aktiviert/deaktiviert.



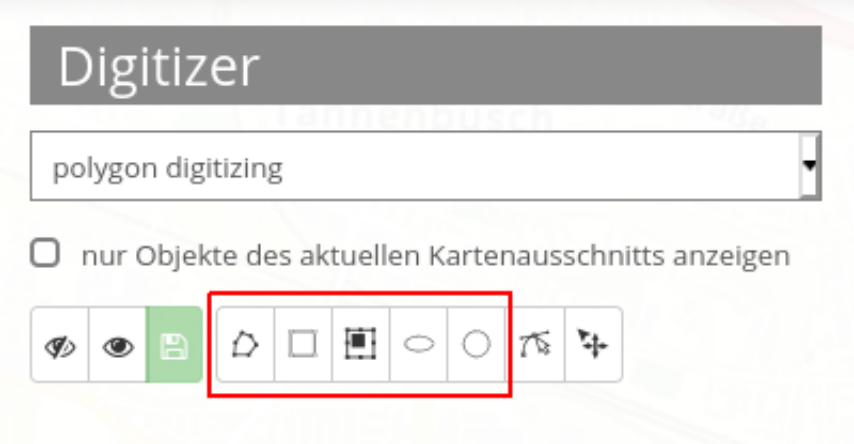
Linien

Durch einen Klick auf den Button *"Linie erstellen"* wird die Funktion aktiviert/deaktiviert.



Polygone

Es können unterschiedliche Arten von Polygonen erstellt werden. Dies ist über eine Aktivierung des jeweiligen Buttons für Flächen, Rechtecke, Enklaven, Ellipsen oder Kreise möglich.



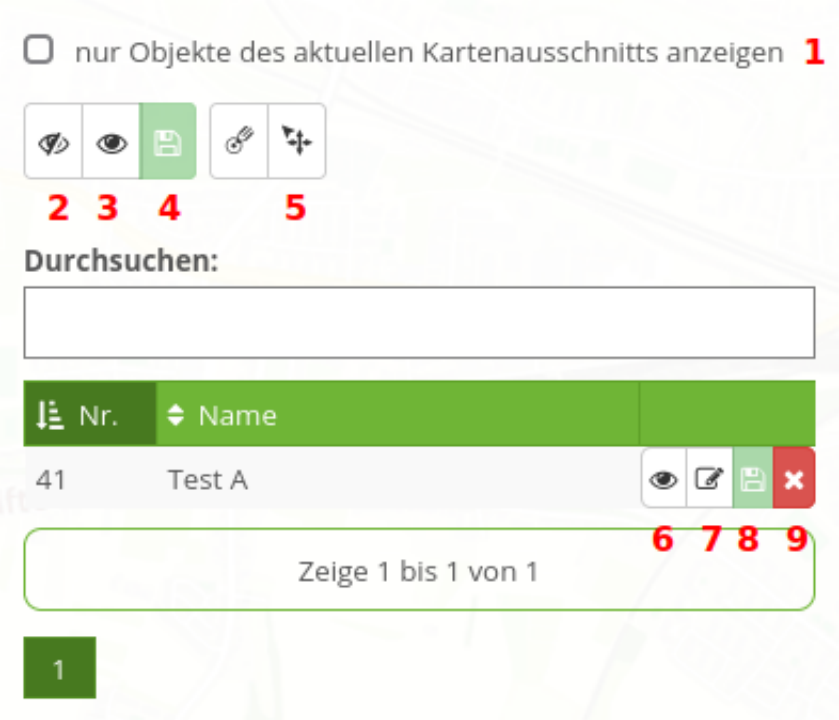
Mithilfe der Maus kann nun die zuvor ausgewählte Geometrieart in der Karte erstellt werden. Anschließend öffnet sich ein Pop-up Fenster, welches das vordefinierte Sachdatenformular gemäß der Yaml-Konfiguration ausgibt und die Erfassung dieser zulässt.

Geometrien bearbeiten, speichern oder löschen

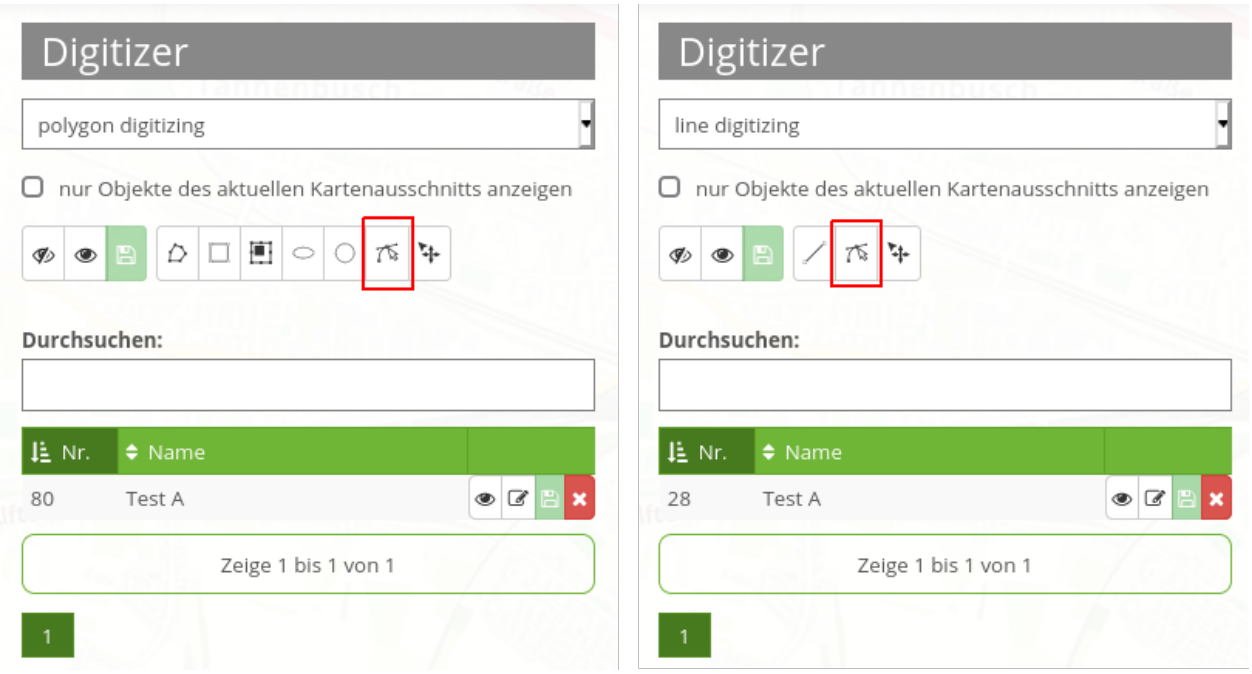
Die Speicherung der Geometrien erfolgt in der jeweils definierten Datenbanktabelle. Die Objekte werden in der Karte angezeigt und darüber hinaus im Digitizer-Element in Form einer Tabelle aufgelistet. Dies erleichtert die Verwaltung der Geometrien. In der Tabelle können beliebige Spalten ausgegeben werden. Im Beispiel werden die Nummer (ID wird automatisch erzeugt) sowie Name jedes Objekts angezeigt. Es besteht die Möglichkeit, die Sortierung der Spalten zu ändern sowie die Tabelle zu durchsuchen.

In der Auflistung können nur Objekte des aktuellen Kartenausschnitts angezeigt werden **1**. Weiterhin besteht die Option, alle Objekte auszublenden **2** oder einzublenden **3**. Änderungen können für alle Objekte gespeichert werden **4**. Eine Bearbeitung wäre z.B. die Verschiebung von Objekten **5**. Nach Aktivierung dieses Buttons können Geometrien mithilfe der Maus verändert werden.

Es besteht weiterhin die Möglichkeit, nur einzelne Objekte auszublenden **6** sowie deren Modifikationen individuell abzuspeichern **8**. Veränderungen der Sachdaten sind ebenfalls möglich **7**. Zuletzt kann jedes Objekt gelöscht werden **9**.



Die zuvor beschriebenen Funktionen sind identisch bei allen Geometrien. Es gibt außerdem noch die Option, Linien und Polygone zu modifizieren. Mit dieser Funktion können Stützpunkte eingefügt oder Eckpunkte verschoben werden. Durch Klick auf den Button *"Bearbeiten"* wird die Funktion aktiviert. Um ein Objekt nun zu modifizieren, muss dieses nun individuell durch einen Klick ausgewählt werden.



3.2.2 Data Manager

Der Data Manager ähnelt in seiner Funktionalität dem *Digitizer*. Allerdings speichert das Element nur Sachdaten und keine Geodaten. Es kann zur Pflege von Referenztabellen verwendet werden, beispielsweise einer Tabelle mit Kontaktinformationen.

Konfiguration

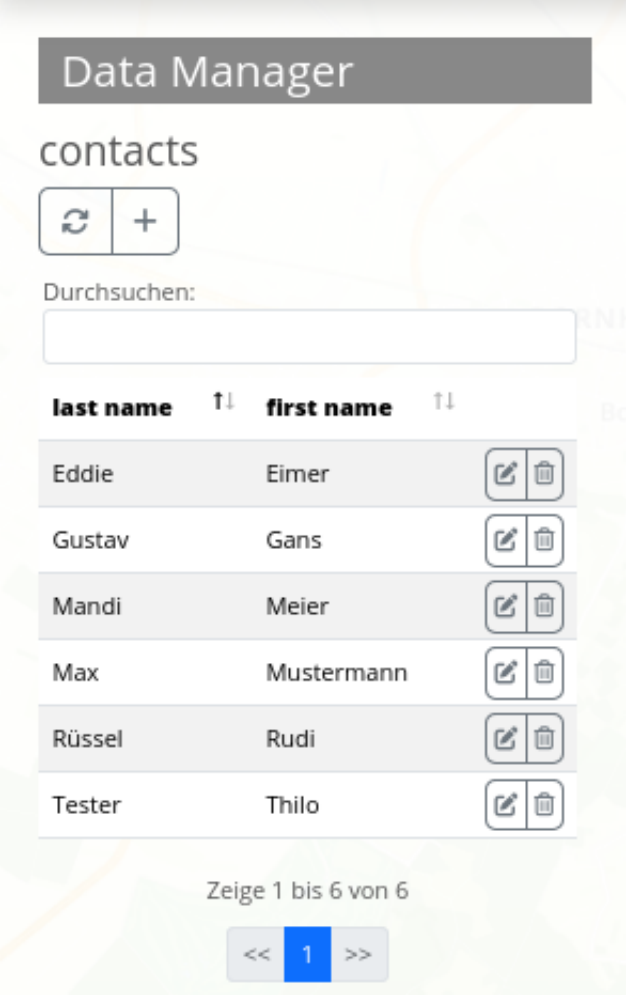
Der Data Manager benötigt einen Zugriff auf die Datenbank, in der die zu editierenden Tabellen liegen. Sie müssen dazu einen Datenbankzugriff konfigurieren. Mehr zu diesem Thema finden Sie unter *Konfiguration der Datenbank*.

Konfigurationsbeispiel

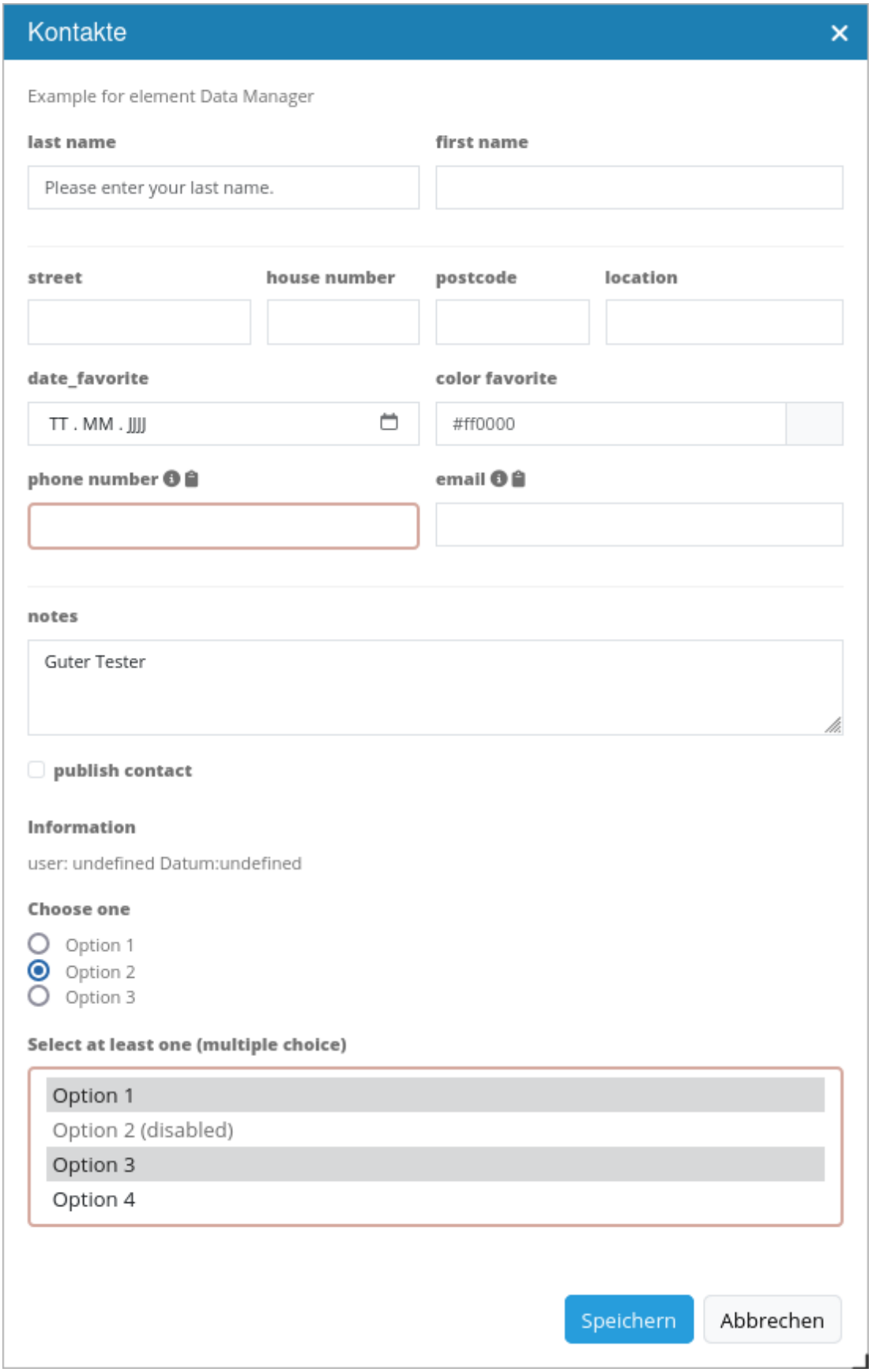
Die Definition des Data Managers erfolgt in YAML-Syntax. Hier definieren Sie die Datenbankverbindung, die editierbaren Felder und den Formularaufbau.

Hint: Bei fehlerhaften Angaben zur Datenbank, Feldern und Formularfehlern erscheinen Fehlermeldungen. Bei Produktivumgebungen erscheint eine allgemeine Fehlermeldung. Falls Sie die detaillierte Fehlermeldung sehen möchten, sollten Sie die Anwendung als Entwicklungsumgebung aufrufen. Weitergehende Informationen unter *Starten von Mapbender als Produktivumgebung*.

Der Data Manager kann beispielsweise genutzt werden, um Kontaktinformationen zu speichern.



In dem dargestellten Konfigurationsbeispiel öffnet sich eine Eingabevorlage nach Klick auf + .


Kontakte
Example for element Data Manager
last name
Please enter your last name.
first name
street
house number
postcode
location
date_favorite
TT . MM . JJJJ
color favorite
#ff0000
phone number
email
notes
Guter Tester
publish contact
Information
user: undefined Datum:undefined
Choose one
Option 1
Option 2
Option 3
Select at least one (multiple choice)
Option 1
Option 2 (disabled)
Option 3
Option 4
Speichern
Abbrechen

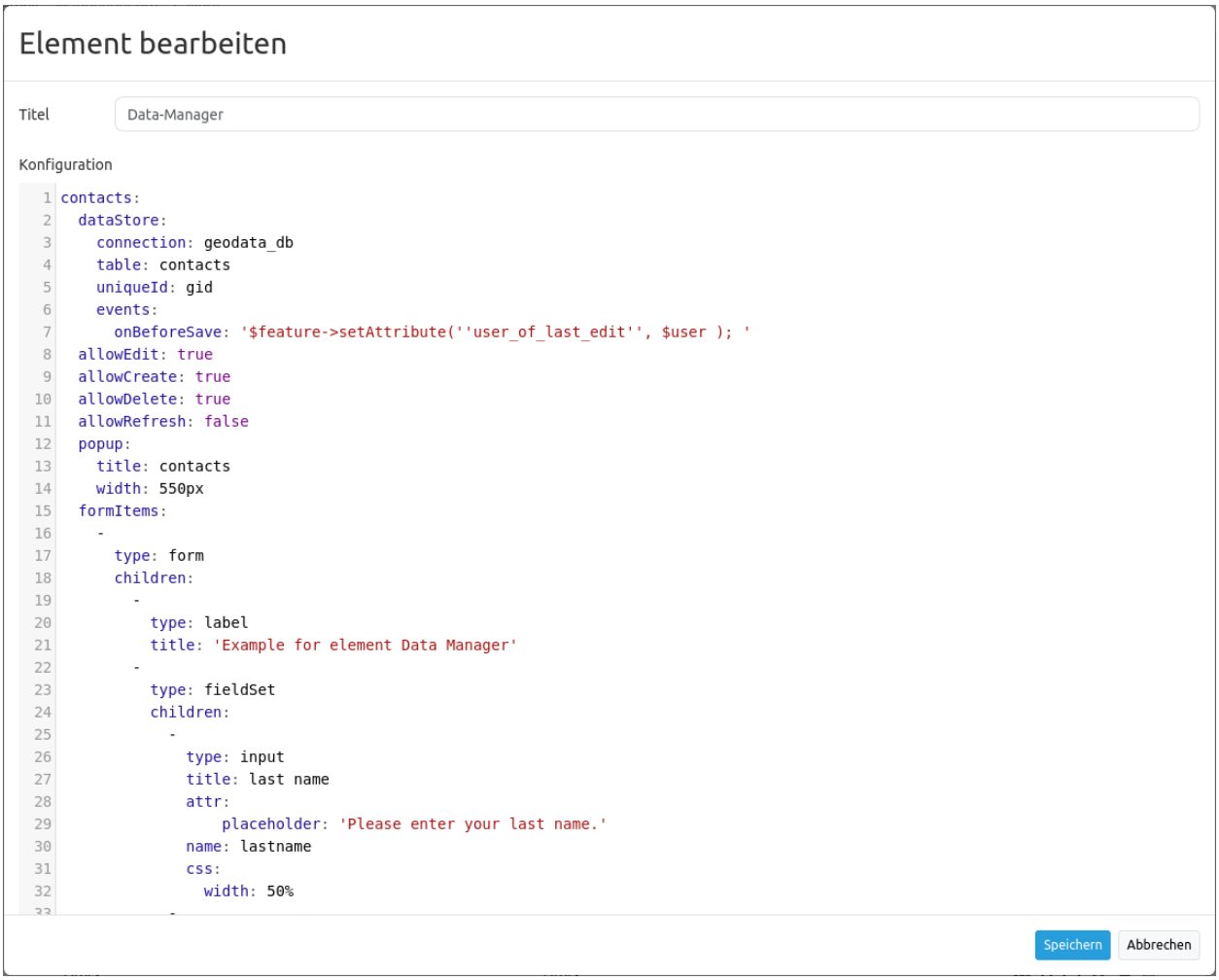
SQL for the contact table

Die nachfolgenden SQL-Befehle müssen in Ihrer Datenbankumgebung ausgeführt werden, um die Kontakttabelle aus dem Beispiel zu erstellen.

```
CREATE TABLE public.contacts (
     gid serial PRIMARY KEY,
     firstname varchar,
     lastname varchar,
     street varchar,
     housenumber varchar,
     postcode integer,
     location varchar,
     phone varchar,
     email varchar,
     notes varchar,
     public boolean,
     date_of_creation date DEFAULT date(now()),
     user_of_last_edit varchar,
     date_favorite date,
     color_favorite varchar
 );
```

Element-Konfiguration

Das Element kann danach in die Sidepane eingebettet werden.



YAML-Definition

Im Folgenden ist der YAML-Code des oben dargestellten Konfigurationsbeispiels hinterlegt:

```yaml
contacts:
  dataStore:
    connection: geodata_db
    table: contacts
    uniqueId: gid
    events:
      onBeforeSave: '$feature->setAttribute(''user_of_last_edit'', $user ); '
  allowEdit: true
  allowCreate: true
  allowDelete: true
  allowRefresh: false
  popup:
    title: contacts
    width: 550px
  formItems:
    -
      type: form
      children:
```

(continues on next page)

(continued from previous page)

```
    -
      type: label
      title: 'Example for element Data Manager'
    -
      type: fieldSet
      children:
        -
          type: input
          title: last name
          attr:
              placeholder: 'Please enter your last name.'
          name: lastname
          css:
            width: 50%
        -
          type: input
          title: 'first name'
          name: firstname
          css:
            width: 50%
    -
      type: breakLine
    -
      type: fieldSet
      children:
        -
          type: input
          title: street
          name: street
          css:
            width: 30%
        -
          type: input
          title: 'house number'
          name: housenumber
          css:
            width: 20%
        -
          type: input
          title: postcode
          name: postcode
          css:
            width: 20%
        -
          type: input
          title: location
          name: location
          css:
            width: 30%

    -
      type: fieldSet
      children:
        -
          type: date
          title: date_favorite
```

(continues on next page)

(continued from previous page)

```
        name: date_favorite
        css:
          width: 50%
      -
        type: colorPicker
        title: color favorite
        name: color_favorite
        attr:
          placeholder: '#ff0000'
        css:
          width: 50%
    #-
    #  type: breakLine
    -
      type: fieldSet
      children:
        -
          type: input
          title: phone number
          name: phone
          mandatory: true
          mandatoryText: 'Please add your phone number.'
          infoText: 'Help: Please add your phone number.'
          copyClipboard: true
          css:
            width: 50%
        -
          type: input
          title: email
          infoText: 'Help: Please add your e-mail.'
          copyClipboard: true
          placeholder: 'Please enter your e-mail.'
          name: email
          css:
            width: 50%
    -
      type: breakLine
    -
      type: textArea
      title: notes
      placeholder: 'You can leave notes here.'
      name: notes
      value: 'Guter Tester'
    -
      type: checkbox
      name: public
      value: true
      title: 'publish contact'
    -
      type: text
      title: 'Information'
      text: "'user: ' + data.user_of_last_edit + ' Datum:' + data.date_of_creation\n"
    - type: radioGroup
      title: Choose one
      name: choice_column_1
      options:
```

(continues on next page)

(continued from previous page)

```
          - label: Option 1
            value: v1
          - label: Option 2
            value: v2
          - label: Option 3
            value: v3
        value: v2   # Pre-select second option by default for new items
      - type: select
        title: Select at least one (multiple choice)
        attr:
          required: required
          multiple: multiple
        name: choice_column_2
        options:
          - label: Option 1
            value: v1
          - label: Option 2 (disabled)
            value: v2
            attr:
              disabled: disabled
          - label: Option 3
            value: v3
          - label: Option 4
            value: v4
        value: v1,v3   # use comma-separated values for default multi-select value
table:
  autoWidth: false
  columns:
    -
      data: lastname
      title: last name
    -
      data: firstname
      title: first name
  info: true
  lenghtChange: false
  ordering: true
  pageLength: 10
  paging: true
  processing: true
  searching: true
```

3.2.3 Skizzen (Sketch)

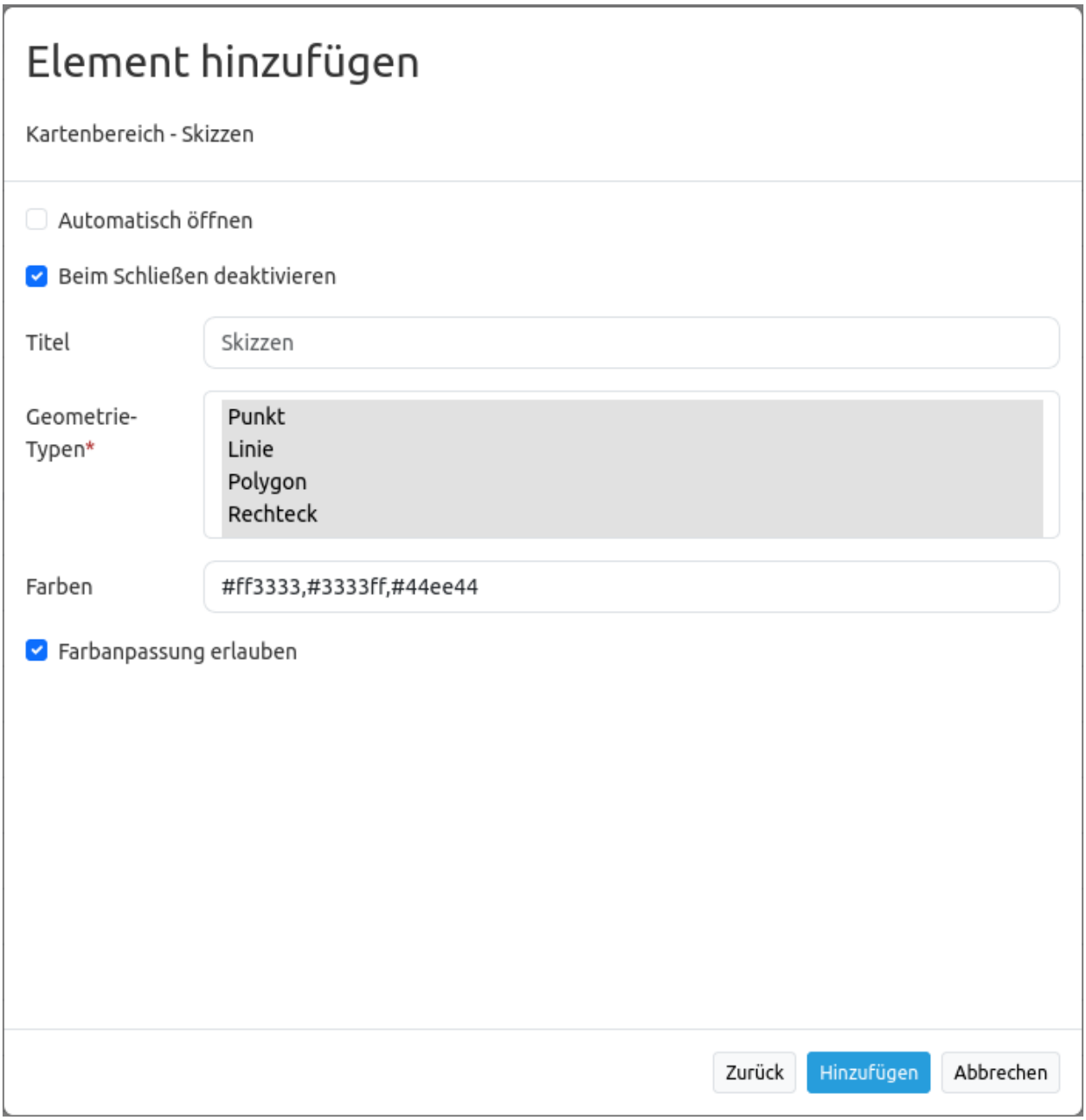
Mit diesem Element können temporär verfügbare Objekte in der Karte erstellt werden. Diese werden nicht in einer Datenbank gespeichert und gehen beim browsergesteuerten Aktualisieren oder Schließen der Anwendung verloren. Erstellte Skizzen werden im *Bildexport (ImageExport)* und *Druck (PrintClient)* ausgegeben.

Mit Skizzen können folgende Geometrietypen erstellt werden:

- Punkt
- Linie
- Polygon

- Rechteck
- Kreis
- Text

Konfiguration



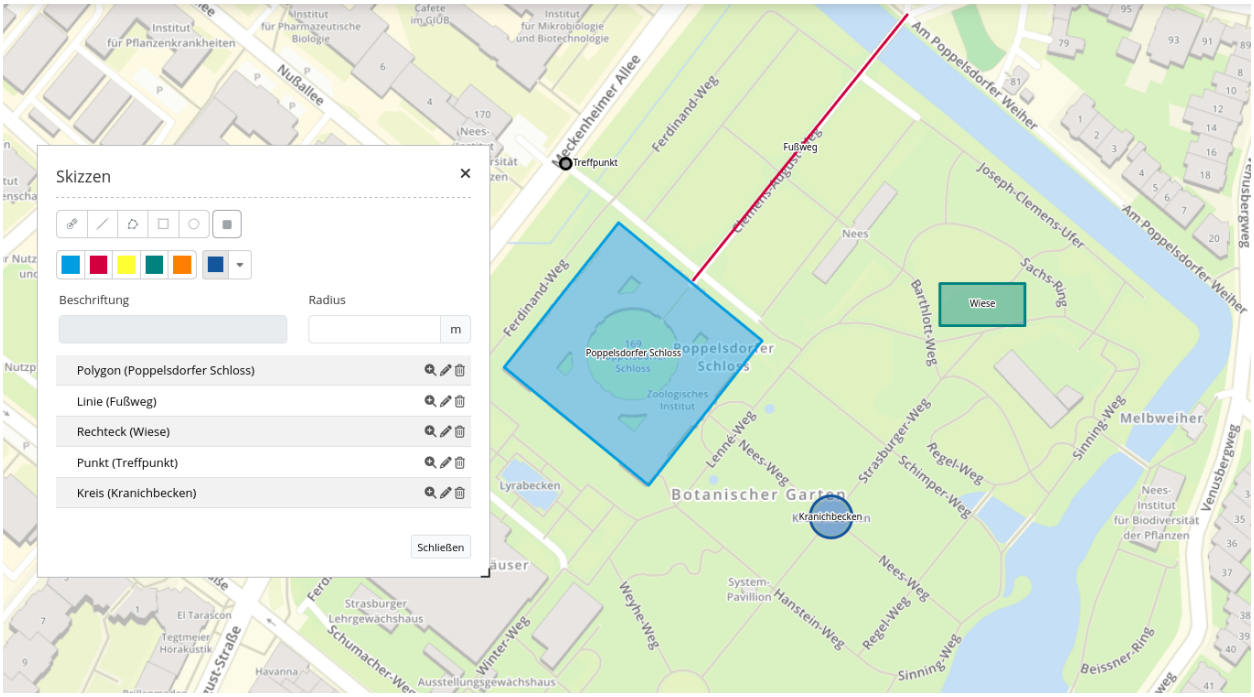
- **Automatisch öffnen:** Das Element wird beim Anwendungsstart automatisch geöffnet (Standard: deaktiviert).
- **Beim Schließen deaktivieren:** Erstellte Skizzen werden nach Schließen des Elements (Button oder Sidepane) nicht mehr in der Karte angezeigt. Erneutes Öffnen zeigt erstellte Skizzen wieder an.

- **Titel:** Titel des Elements. Dieser wird in der Layouts Liste angezeigt und ermöglicht, mehrere Button-Elemente voneinander zu unterscheiden. Der Titel wird außerdem neben dem Button angezeigt, wenn *Beschriftung anzeigen* aktiviert ist.
- **Geometrietypen:** Liste der unterstützten Geometrietypen (Pflichtfeld).
- **Farben:** Texteingabefeld für optionale Farbanpassung (Hex-Codes). Kommaseparierte Eingabe. Benötigt mindestens einen Farbcode.
- **Farbanpassung erlauben:** Aktiviert einen Button als Farbwähler, mit dem aus der Anwendung heraus Skizzen farblich verändert werden können.

Geometrien erstellen

- **Punkt zeichnen:** wird über den Button “Punkt” aktiviert. Punkte werden über einen Einfachklick in die Karte definiert.
- **Linie zeichnen:** wird über den Button “Linie” aktiviert. Liniensegmente werden mit einem Einfachklick definiert. Das Abschließen der Zeichnung erfolgt mit einem Doppelklick.
- **Polygon zeichnen:** wird über den Button “Polygon” aktiviert. Eckpunkte werden mit einem Einfachklick definiert. Das Abschließen der Zeichnung erfolgt mit einem Doppelklick.
- **Rechteck zeichnen:** wird über den Button “Rechteck” aktiviert. Rechtecke werden nach einem Einfachklick mit der Maus aufgezogen und mit einem erneuten Einfachklick erstellt.
- **Kreis zeichnen:** wird über den Button “Kreis” aktiviert. Kreise werden nach einem Einfachklick mit der Maus aufgezogen und mit einem erneuten Einfachklick erstellt. Nach dem Zeichnen eines Kreises kann dessen Radius über die Oberfläche angepasst werden.
- **Text setzen:** wird über den Button “Text” aktiviert. Damit eine Textsetzung erfolgen kann, muss der Text zunächst unter “Beschriftung” definiert werden.

Beispiele für die verschiedenen Skizzengeometrien zeigt die Abbildung:



Löschen, Bearbeiten und Zoomen

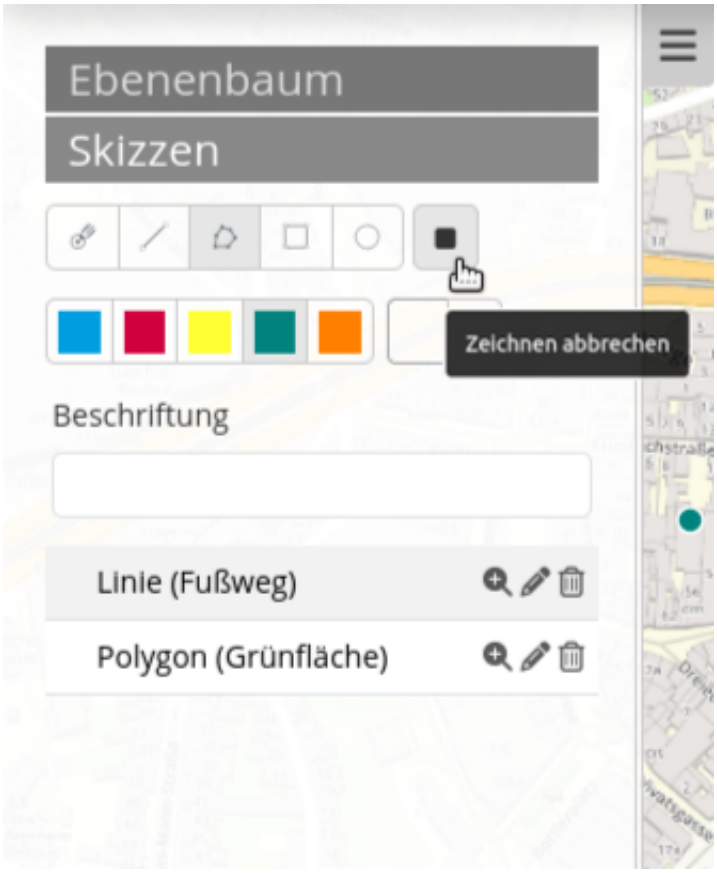
Gezeichnete Skizzen können individuell gelöscht [1] oder bearbeitet [2] werden. Außerdem besteht die Möglichkeit, auf einzelne Geometrien zu zoomen [3].

Farbauswahl ändern

Nach der Konfiguration im *Backend* finden sich Farben als Button im Skizzen-Werkzeug. Um einer Skizze eine bestimmte Farbe zuzuweisen, muss zunächst das gewünschte Geometriewerkzeug und anschließend der gewünschte Farbbutton angeklickt werden. Auch bereits gespeicherte Geometrien lassen sich farblich anpassen: Dazu muss eine Skizze per Klick auf "Editieren der Geometrie" (Bearbeiten-Button) ausgewählt und anschließend die gewünschte Farbe per Mausklick zugewiesen werden. Ein Farbwechsel per Klick auf den entsprechenden Farb-Button ist auch während des Erstellens einer Geometrie möglich. Der Farbwähler, der im *Backend* mit **Farbanpassung erlauben** aktiviert werden kann, erlaubt eine freie Farbwahl auf Anwendungsebene. Sofern das Auswählen von Farben komplett deaktiviert und stattdessen nur eine einzige Farbe angeboten werden soll, muss die gewünschte Farbe im Konfigurationsdialog definiert und die Checkbox **Farbanpassung erlauben** deaktiviert werden.

Zeichnen abbrechen

Das Zeichnen von Skizzen kann mithilfe von "Zeichnen abbrechen" (Stop-Button) unterbrochen werden.



YAML-Definition

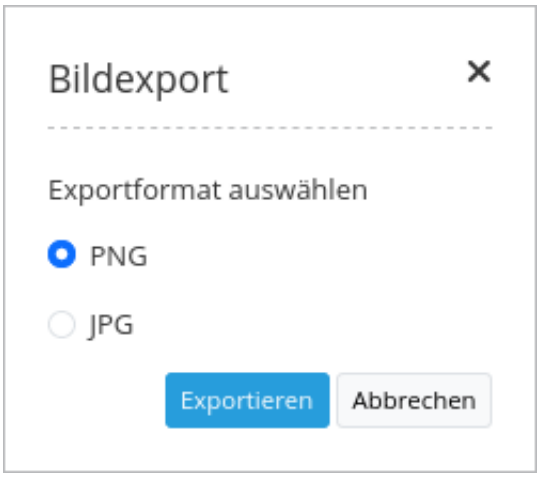
Diese Vorlage kann genutzt werden, um das Element in einer YAML-Applikation einzubinden.

```
tooltip: 'Sketch'                # Text des Tooltips
target: ~                        # ID des Kartenelements
auto_activate: false             # (Standard: false)
deactivate_on_close: true        # Gezeichnete Geometrien werden durch Schließen des Elements
↪deaktiviert (Standard: true)
geometrytypes: point,line,polygon,rectangle,circle,text   # Liste der unterstützten Geometrietypen
colors:                          # Definition optionaler CSS-Farben in hexadezimaler Notation möglich
  - '#ff3333'                    # Standardfarbe rot
  - '#3333ff'                    # Standardfarbe blau
  - '#44ee44'                    # Standardfarbe grün
allow_custom_color: true         # Aktiviert Farbauswahl im Skizzen-Element (Standard: true)
```

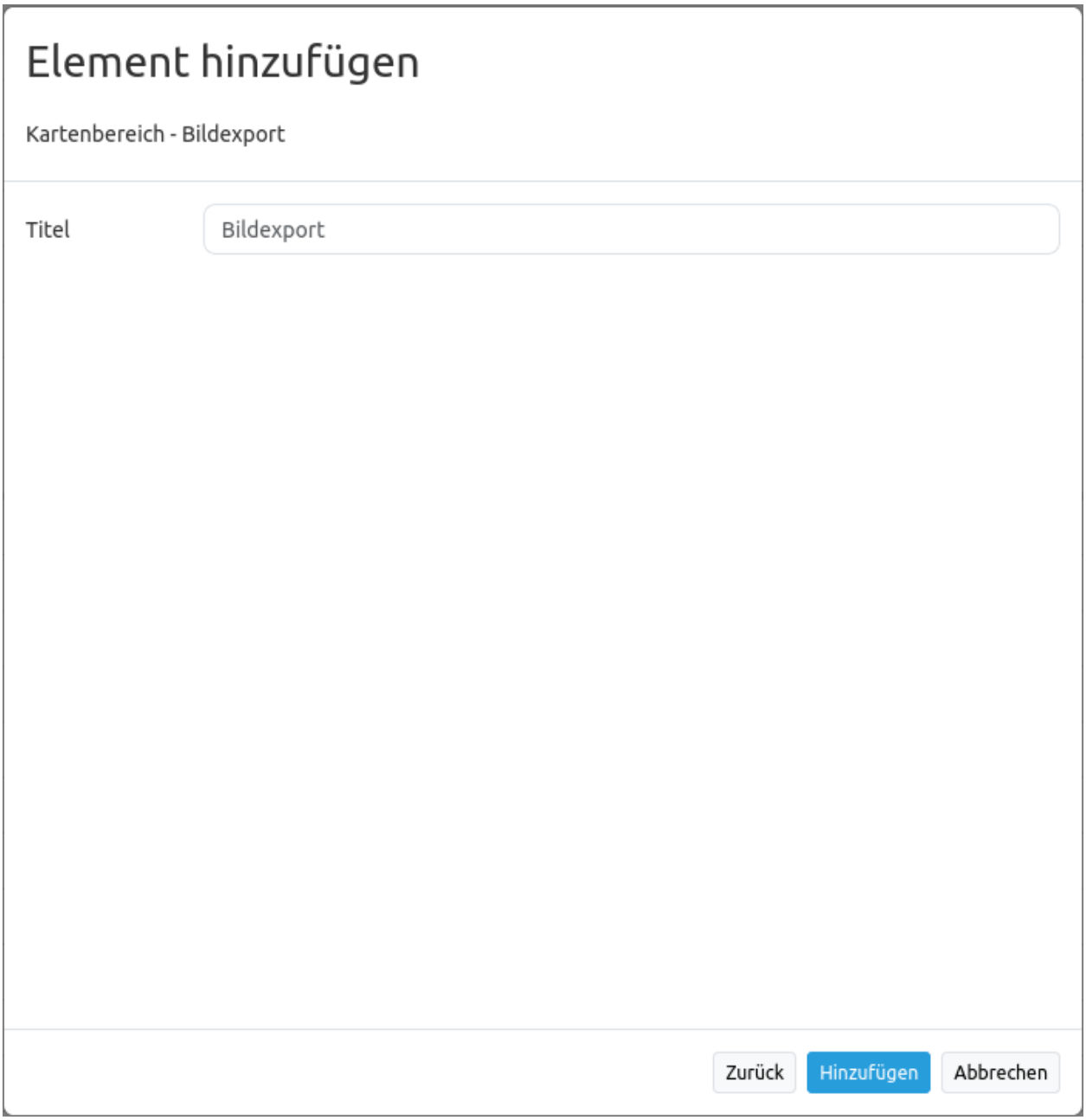
3.3 Export / Druck

3.3.1 Bildexport (ImageExport)

Mit dem Bildexport kann die aktuelle Kartenansicht exportiert werden. Das Bild kann als .png oder .jpg abgespeichert werden.



Konfiguration



- **Titel:** Titel des Elements. Dieser wird in der Layouts-Liste angezeigt und ermöglicht die Unterscheidung mehrerer Button-Elemente. Der Titel wird außerdem neben dem Button angezeigt, wenn *Beschriftung anzeigen* aktiviert ist.

YAML-Definition

Diese Vorlage kann genutzt werden, um das Element in einer YAML-Anwendung einzubinden.

```
class: Mapbender\PrintBundle\Element\ImageExport
target: map                        # ID des Kartenelements (z.B. map)
```

Für das Element wird ein Button verwendet. Zu der Konfiguration des Buttons besuchen Sie die Dokumentationsseite unter *Button*.

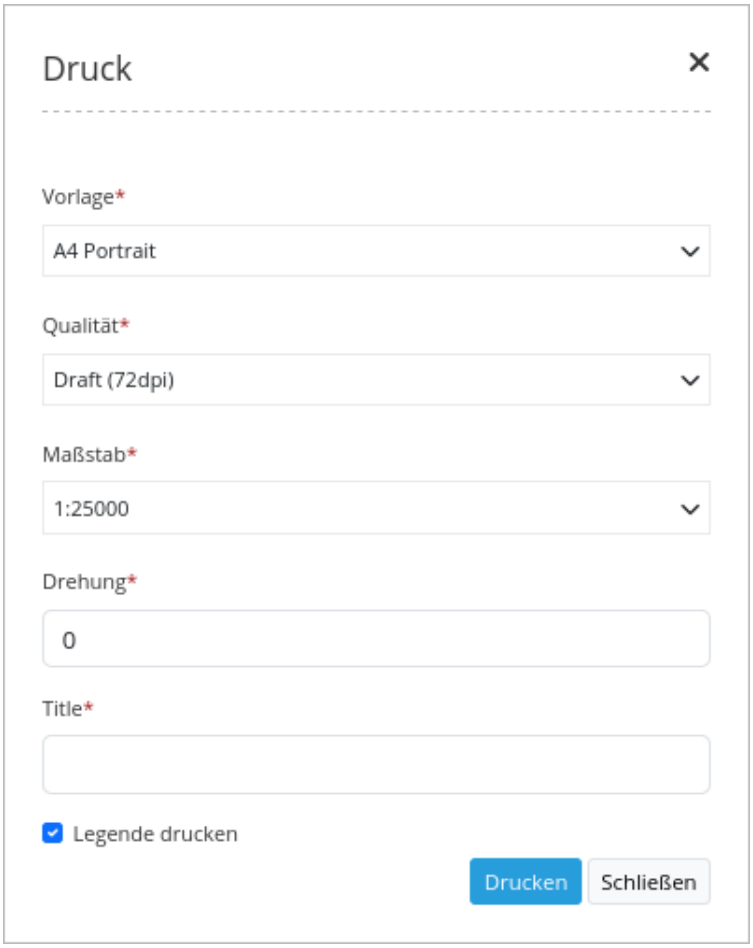
3.3.2 Druck (PrintClient)

Das Druckelement ermöglicht den Druck eines auswählbaren Kartenausschnitts. Diese Dokumentation bietet einen Überblick über dessen Einrichtung und Konfiguration, skizziert die Erstellung individueller Druckvorlagen und erläutert den Druckvorgang sowie seine Konfigurationsoptionen.

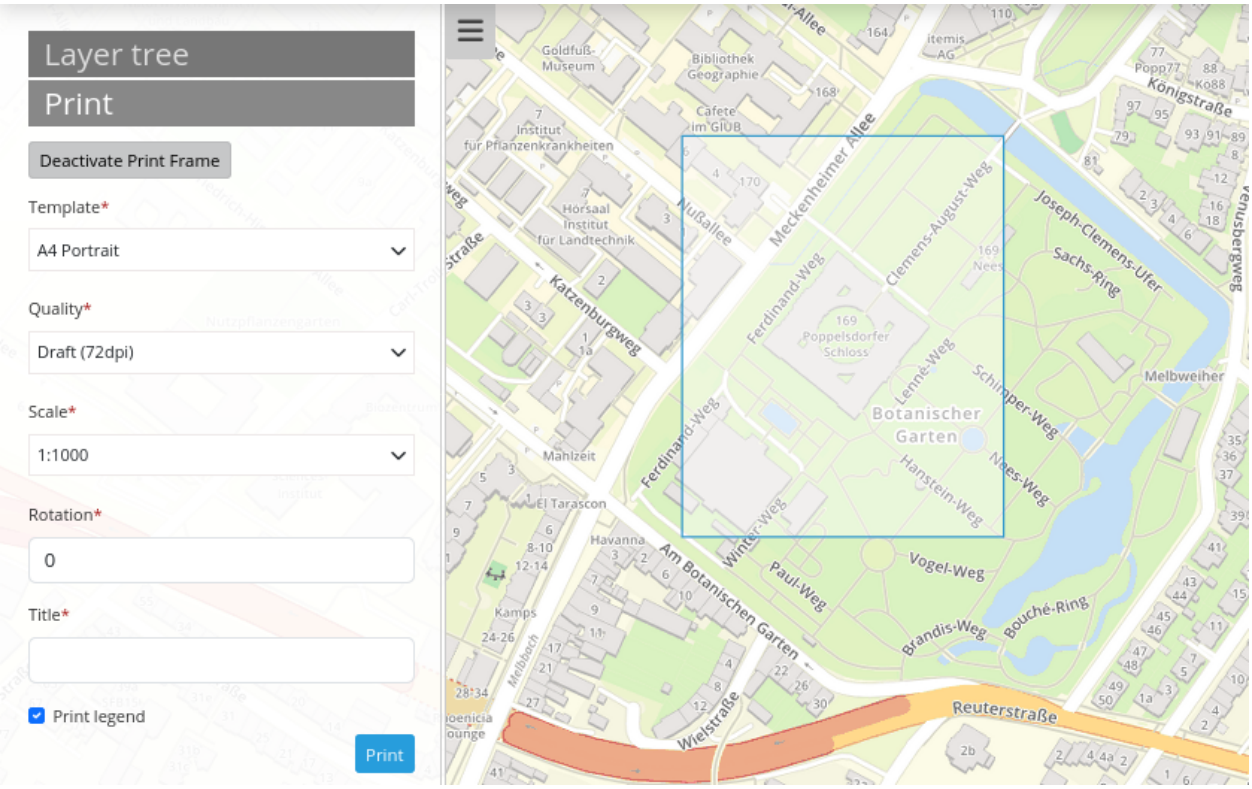
Allgemeines

Mit dem PrintClient können folgende Druckeinstellungen beeinflusst werden:

- Vorlage
- Qualität
- Maßstab
- Drehung
- Titel
- Checkbox *Legende drucken*

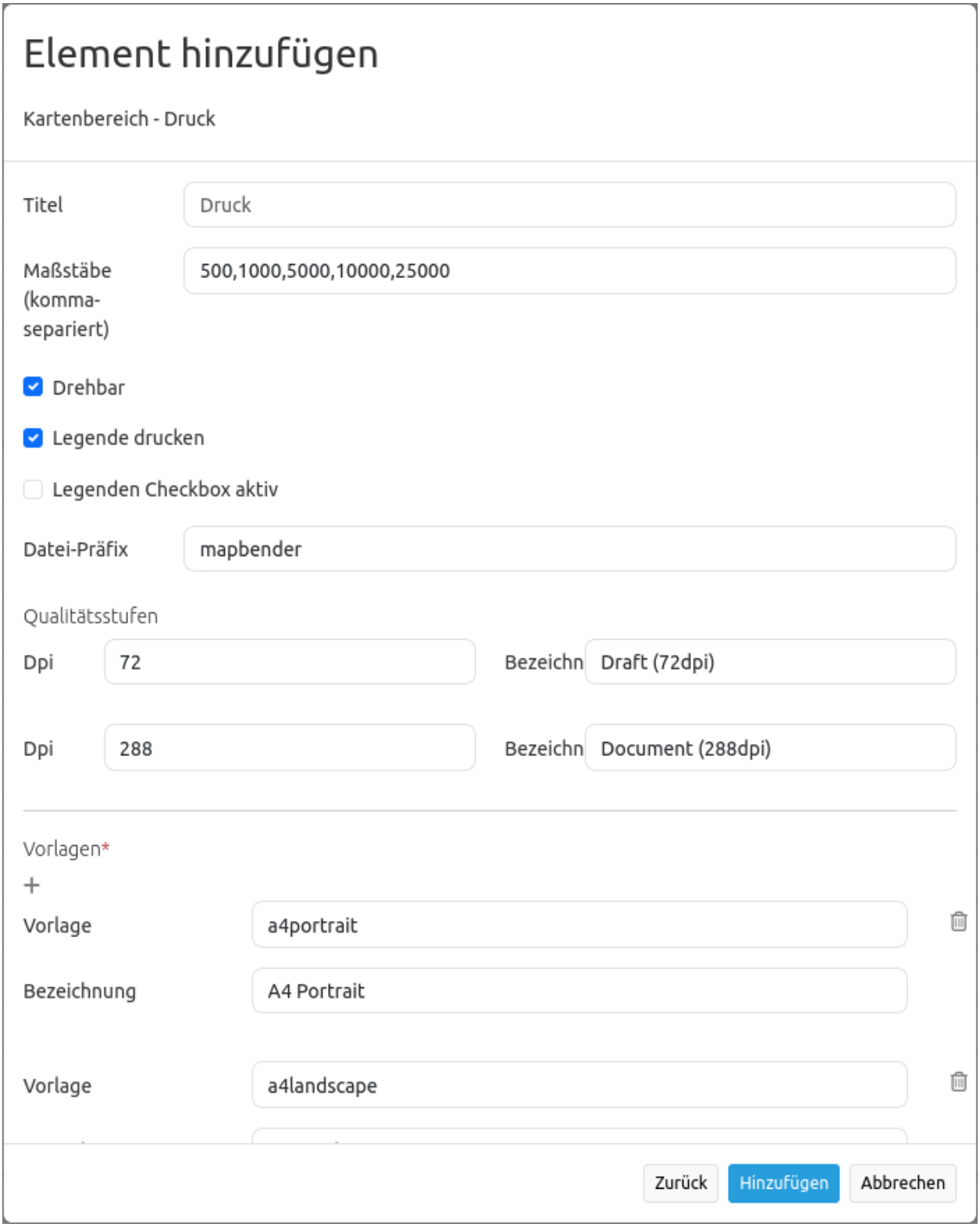


Das Element kann über einen Button aufgerufen oder in der Sidepane als Element eingebunden werden. Sobald der PrintClient angesteuert wird, erscheint auf der Karte ein Druckrahmen. Dieser bestimmt den zu druckenden Bereich der Karte und kann vom Anwender beliebig ausgerichtet werden. Nach dem Druck muss der Druckrahmen wieder deaktiviert werden, damit die Karte wie gehabt genutzt werden kann (im Dialog geschieht dies alles durch das Öffnen und Schließen des Dialogfensters).



Konfiguration

Der PrintClient wird im *Backend* konfiguriert. Er greift dabei auf Druckvorlagen (Templates) zurück. Diese LibreOffice Draw Dateien liegen in den Formaten A0 bis A4 vor. In ihnen können Bereiche für Datum, Maßstab, Nordpfeil, Übersichtskarte und dynamische Bild- und Textbereiche definiert werden.



- **Titel**: Titel des Elements. Dieser wird in der Layouts Liste angezeigt und ermöglicht, mehrere Button-Elemente voneinander zu unterscheiden. Der Titel wird außerdem neben dem Button angezeigt, wenn "Beschriftung anzeigen" aktiviert ist.
- **Maßstäbe (csv)**: Maßstäbe, die in der Selectbox ausgewählt werden können. Wenn keine Maßstäbe

angegeben werden, muss ein beliebiger Maßstab hinzugefügt werden.

- **Drehbar**: definiert, ob der Ausdruck gedreht werden kann (Standard: true).
- **Legende drucken**: fügt eine Checkbox hinzu, die den Druck der Kartenlegende ermöglicht (Standard: false).
- **Legenden Checkbox aktiv**: definiert, ob die "Legende drucken" - Checkbox bei Anwendungsstart aktiv ist
- **Datei-Präfix**: Anpassung des Dateinamens für das PDF. Dieser setzt sich aus dem definierbarem Prefix und der Datums- und Uhrzeitangabe des Drucks zusammen.
- **Qualitätsstufen**: Qualität in dpi-Werten definieren und die dazugehörige Beschriftung angegeben

Vorlagen

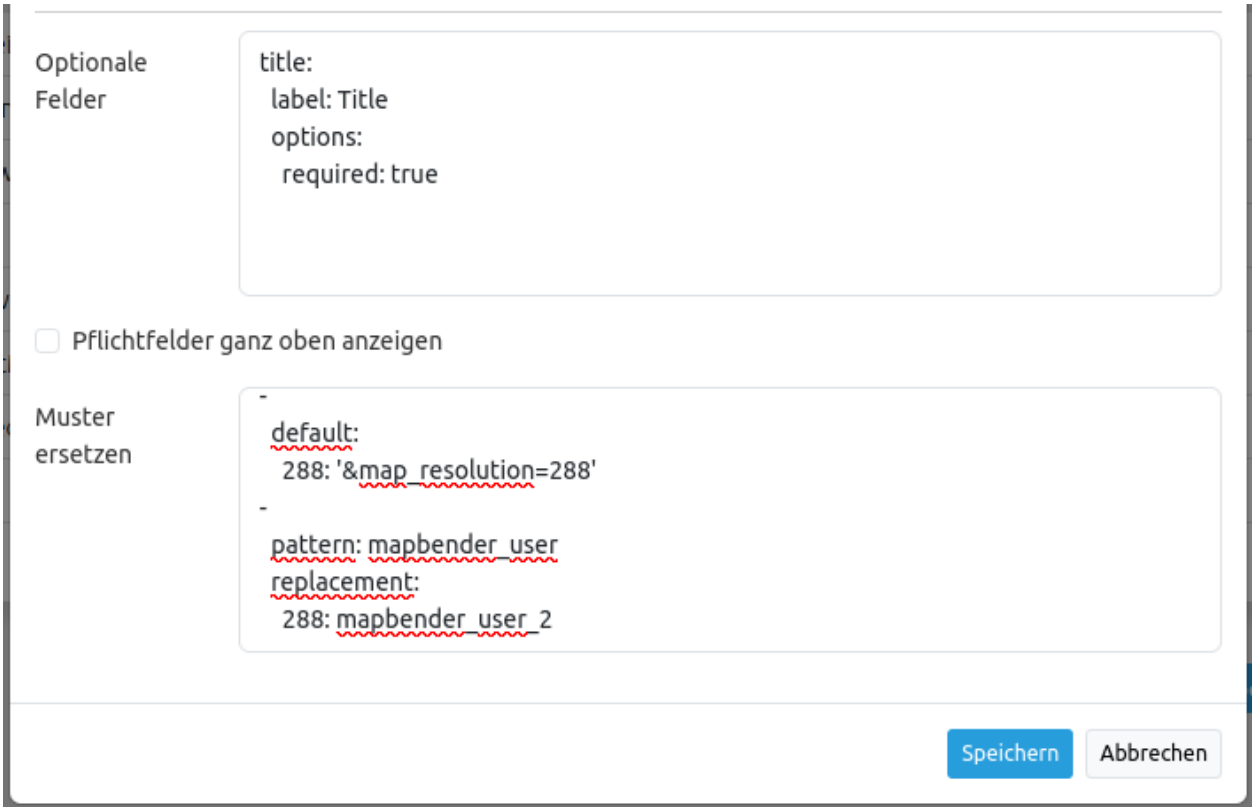
- **Vorlage**: Verweis auf die Druckvorlage (z.B. a4portrait). Löschen und Hinzufügen von Vorlagen ist über die "+"- bzw. "x"-Symbole möglich.
- **Bezeichnung**: Bezeichnung der Druckvorlagen im Frontend (z.B. Portrait A4).

Optionale Felder

Über die Konfiguration folgender Werte können optionale Felder im Druckdialog ermöglicht werden. Eine Beispielkonfiguration mit vier Feldern (Titel, zwei Kommentarfelder, Bearbeiter) gibt die YAML-Definition.

- **title**: Name des optionalen Feldes, der Standardwert ist null (keine optionalen Felder sind definiert).
- **label**: Beschriftung des optionalen Feldes.
- **options**: Typ des optionalen Feldes, muss true oder false sein.
- **Pflichtfelder ganz oben anzeigen**: Ist diese Checkbox aktiv, erscheinen Pflichtfelder im Druckdialog ganz oben.
- **Muster ersetzen**: Verändert den Kartenaufruf, Standardwert ist null. Angaben können hinzugefügt oder verändert werden, wie beispielsweise map_resolution (für MapServer).

Im *Backend* finden Sie die Formularfelder im unteren Teil des Druckdialogs (ein ausführliches und kommentiertes Beispiel steht weiter unten im Abschnitt YAML-Definition).



YAML-Definition

Diese Vorlage kann genutzt werden, um das Element in einer YAML-Anwendung einzubinden.

```
target: map                                          # ID des Kartenelements
type: dialog                                         # element oder dialog, default ist dialog
templates:
    - { template: a4portrait, label: A4 Portrait}    # Vorlagen (template): Vorlagename,
↪Vorlagedateiname ohne Dateierweiterung (Mapbender sucht die Datei a4portrait.odg und a4portrait.pdf),
↪die Vorlagedateien befinden sich in config/MapbenderPrintBundle
    - { template: a4landscape, label: A4 Landscape}  # Vorlagebeschriftung im Druckdialog
scales: [5000, 10000, 25000]                         # Maßstäbe definieren, die in der Selectbox
↪ausgewählt werden können. Wenn keine Maßstäbe angegeben werden, kann ein beliebiger Maßstab in einem
↪Textfeld definiert werden.
quality_levels:                                      # die Qualität in dpi definieren und die
↪dazugehörige Beschriftung angegeben
    - { dpi: 72 , label: Draft (72dpi)}              # die erste Angabe ist der dpi Wert, die
↪zweite Angabe ist die Beschriftung
    - { dpi: 288,  label: Document (288dpi)}         # es können weitere dpi-Werte angegeben werden
rotatable: true                                      # true/false, ob der Ausdruck gedreht werden
↪kann (Standard: true).
legend: true                                         # true/false, ob die Legende mitgedruckt
↪werden soll (Standard: false).
legend_default_behaviour: false                      # true/false, Legenden Checkbox standardmäßig
↪ausgewählt
file_prefix: mapbender                               # Definition des Dateinames für das PDF (wird
↪zusammengesetzt zu file_prefix_date.pdf)
optional_fields:                                     # es können optional weitere Felder definiert
↪werden (z.B. Titel-Feld)
```

(continues on next page)

(continued from previous page)

```
    title:                                          # Name des optionalen Feldes, der Standardwert
↪ist null (keine optionalen Felder sind definiert)
        label: Titel                                # Beschriftung des optionalen Feldes
        type: text                                  # Typ des optionalen Feldes
        options:
            required: true                          # erforderlich: true or false
    comment1:
        label: Kommentar 1
        options: { required: false }
    comment2:
        label: Kommentar 2
        options: { required: false }
    bearbeiter:
        label: Bearbeiter
        options: { required: true }
replace_pattern:                                    # Für den Druck kann der Kartenaufruf
↪verändert werden.
        -                                           # Es können zusätzliche Parameter hinzugefügt
↪werden (wie map_resolution für MapServer).
            default: { 288: '&map_resolution=288' }
        -
            pattern: 'stadtplan.xml'                # oder es können für den Druck optimierte
↪Dienste angefordert werden.
            replacement: { 288: 'stadtplan_4.xml' }
```

Verzeichnisse

Der Nordpfeil

- Das Bild des Nordpfeils befindet sich unter **config/MapbenderPrintBundle/images/**.
- Der NOrdpfeil kann durch ein anderes Bild ersetzt werden.

Die Print-Templates

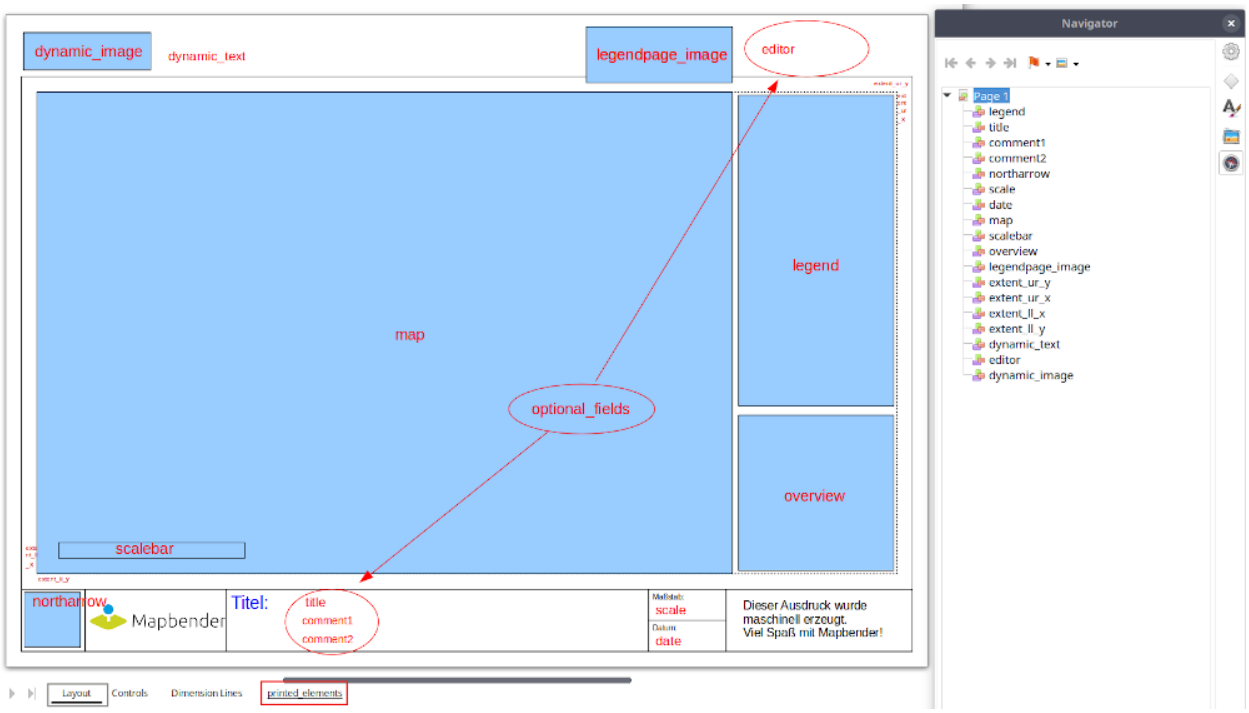
- Die Vorlagen befinden sich unter **config/MapbenderPrintBundle/templates/**.
- Es können eigene Druckvorlagen erstellt und hinzugefügt werden.

Die Druck-Dateien (pdf)

- Mapbender zeigt die generierte PDF direkt im Browser an und stellt sie zum Download bereit.
- Mapbender speichert die Dateien des Warteschleifendrucks hingegen standardmäßig unter **public/prints/**.

Erstellen einer individuellen Vorlage

Um eine individuelle Druckvorlage zu erstellen, kann eine vorhandene Druckvorlage (ODG-Datei, LibreOffice Draw) kopiert und anschließend bearbeitet werden. Die Vorlage kann feste Objekte wie ein Logo, Copyright oder Druckinformationen beinhalten. Zusätzlich muss eine dynamische Ebene für die dynamisch erzeugten Elemente (wie die Karte, die Übersichtskarte, den Nordpfeil, den Maßstab, das Datum und optionale Felder) erstellt werden. Die dynamische Ebene ist eine zusätzliche nicht druckbare Ebene. Eine Ebene in LibreOffice Draw kann folgendermaßen hinzugefügt werden: **Menü: Einfügen -> Ebene... -> Namen für die Ebene definieren und Checkbox "druckbar" deaktivieren**. Die Zuordnung von Elementen zu einer Ebene erfolgt über Drag & Drop.

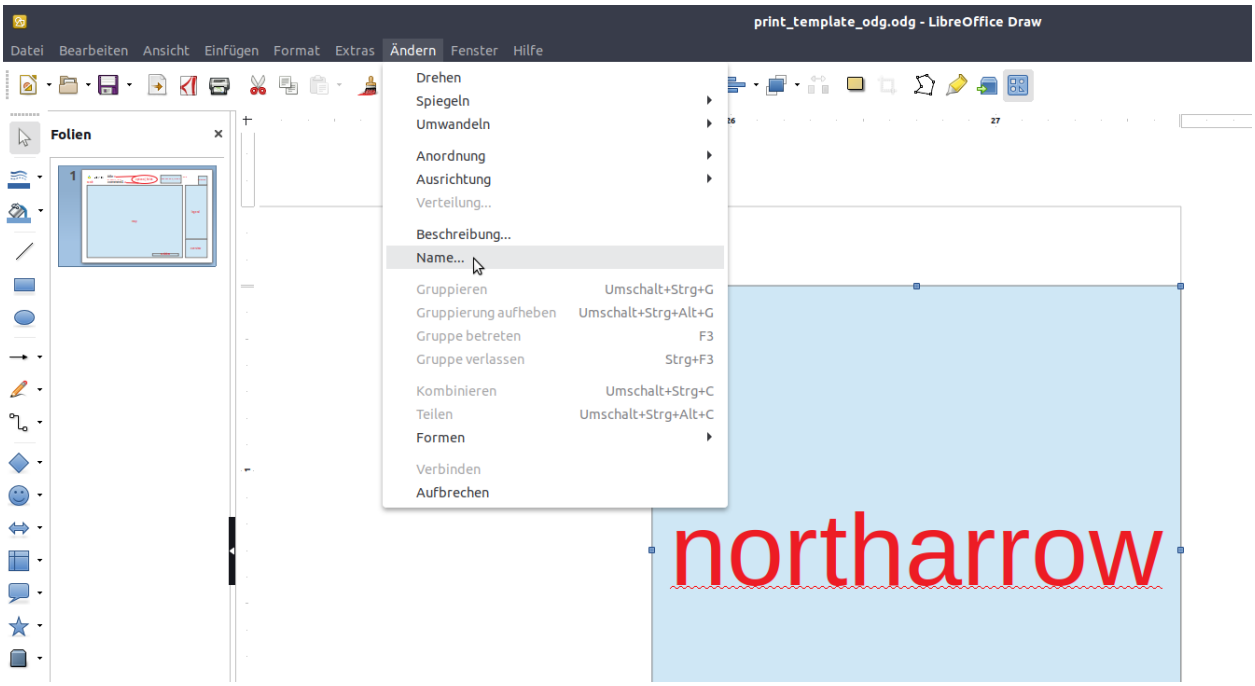


Die folgenden Bereiche liegen standardmäßig in Mapbender vor:

- map (Karte)
- overview (Übersichtskarte)
- scale (Maßstabsangabe in der Form 1:1000)
- scalebar (Maßstabsleiste)
- date (Datum in der Form 10.10.2014)
- northarrow (Nordpfeil)
- extent_ur_y, extent_ur_x, extent_ll_x, extent_ll_y (Koordinaten des Druckbereichs)
- dynamic_image (dynamisches Bild - gekoppelt an die Gruppe)
- dynamic_text (dynamischer Text - gekoppelt an die Gruppe)

Optionale Felder (z.B. Titel, Kommentar, Bearbeiter) können über die Element-Konfiguration definiert werden. Diese müssen im nicht druckbaren Bereich einfügt werden, sodass sie nicht in der Vorlage-PDF ausgegeben werden.

Bei der Erstellung dynamischer Texte müssen die jeweiligen Objekte den passenden Namen haben, z.B. das Kommentarfeld **comment1** muss auch in der Vorlage **comment1** heißen. Zum Benennen der Objektfelder reicht es nicht den Textinhalt anzupassen. Die Felder müssen selektiert und über **Menü: Format** → **Name...** angepasst werden.



Die Vorlage muss als PDF unter dem gleichen Namen wie die ODG-Datei exportiert werden. Namen werden ohne Dateierweiterung in der YML-Druck-Definition verwendet.

Das Druck-Skript liest die Informationen (Position, Größe, Schriftgröße, Ausrichtung) aus der ODG-Datei aus und verwendet ebenfalls das PDF-Dokument mit den festen Objekten. Aus beiden Dateien und den aktuellen Karten wird eine PDF-Druckdatei erstellt.

Druck von Elementen vor dem Kartenbild

Damit beim Druck der Kartenbereich möglichst groß ist und wenig Platz durch weiße oder leere Bereiche verloren geht, können Elemente vor das Kartenbild gelegt werden. Besonders hilfreich ist dies vor allem bei großen Druckformaten wie DIN A1, die einen verhältnismäßig breiten weißen Rand aufweisen.

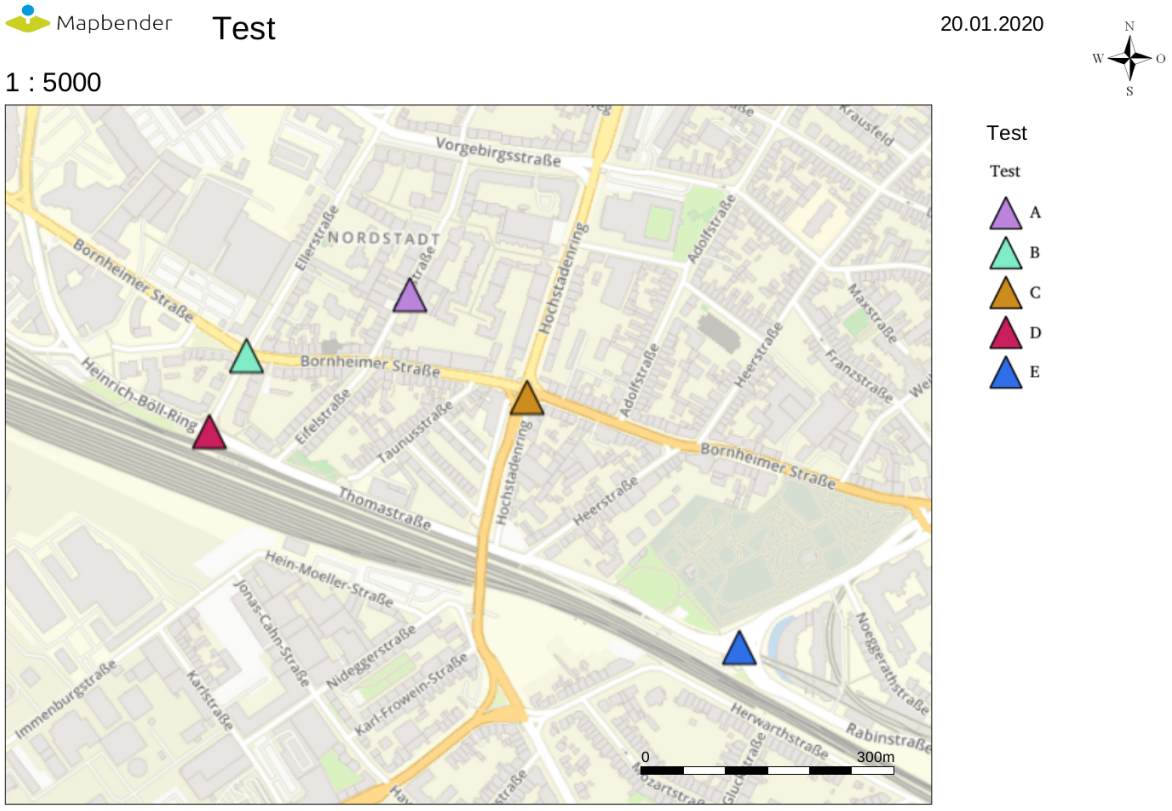
Für die Nutzung dieser Funktion müssen die Templates angepasst und transparente PDF-Vorlagen erzeugt werden.

Templates anpassen:

- **Elemente neu anordnen, am besten vor weißem Hintergrund**
  - Anordnung der Elemente im Vordergrund (Rechtsklick Anordnung Ganz nach vorn)
  - Anordnung Karte = ganz nach hinten (Rechtsklick Anordnung Ganz nach hinten)
- **Alles selektieren**
  - STRG + A drücken
- **Selektion als PDF drucken**
  - Exportieren als PDF
  - Bereich Auswahl statt Alle

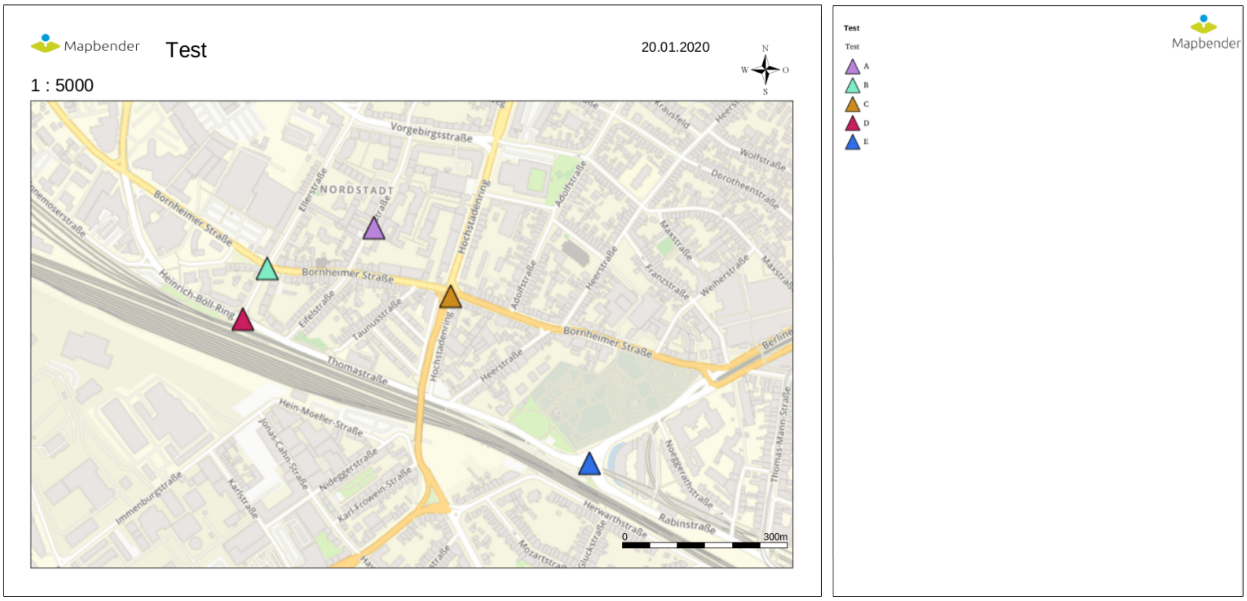
Legende auf der ersten Seite

Neben dem Kartenbereich kann die Legende auf der ersten Seite der Druckvorlage integriert werden. Standardmäßig ist dieser Bereich nicht in den Druckvorlagen vorhanden. Für die Integration der Legende wird die LibreOffice Draw-Druckvorlage angepasst. Hierfür wird ein neues dynamisches Feld mit dem Namen "legend" auf der nicht druckbaren Ebene eingefügt. Die Bezeichnung des Feldes zu "legend" wird über **Menü: Format -> Name...** verändert. Sobald das Element platziert und benannt wurde, wird im Druck die Legende angezeigt. Abschließend muss die ODG-Datei als PDF exportiert und im gleichen Ordner abspeichert werden. Die Legende auf der ersten Seite kann wie folgt aussehen:



Logo auf der Legendenseite

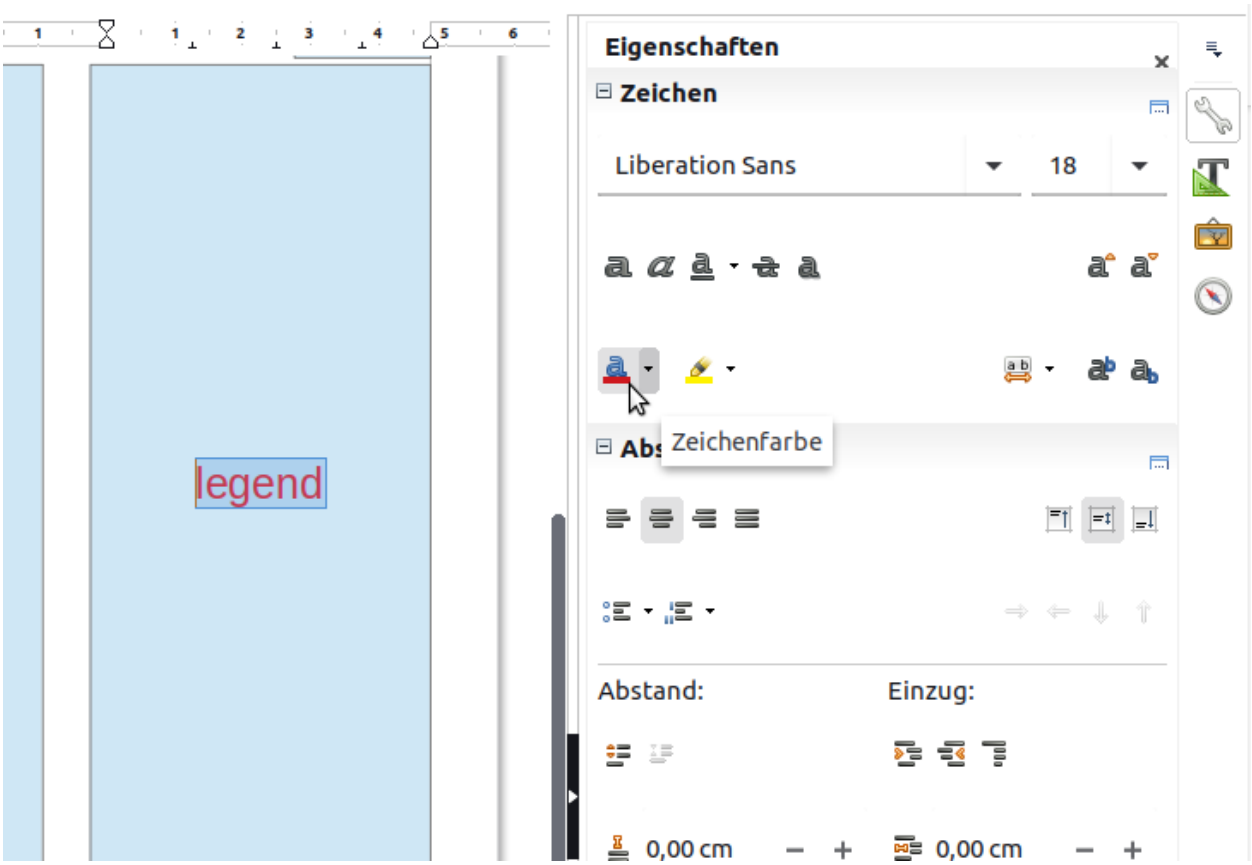
Sofern die Legende weiterhin auf einer zusätzlichen Seite erzeugt werden soll, kann auf der zweiten Seite ebenfalls ein Logo eingebunden werden. Dies erfolgt über das dynamische Element "legendpage_image". Hierfür wird ein neues Feld in der nicht druckbaren Ebene eingefügt, über **Menü: Format -> Name...** in "legendpage_image" umbenannt und an gewünschter Stelle platziert. Das Logo oder Bild wird im Ordner **config/MapbenderPrintBundle/images/** unter dem Namen "legendpage_image.png" abgespeichert. Das erzeugte PDF kann beispielsweise aus folgenden beiden Seiten bestehen:



Auf der zweiten Seite erscheint das eingefügte Logo.

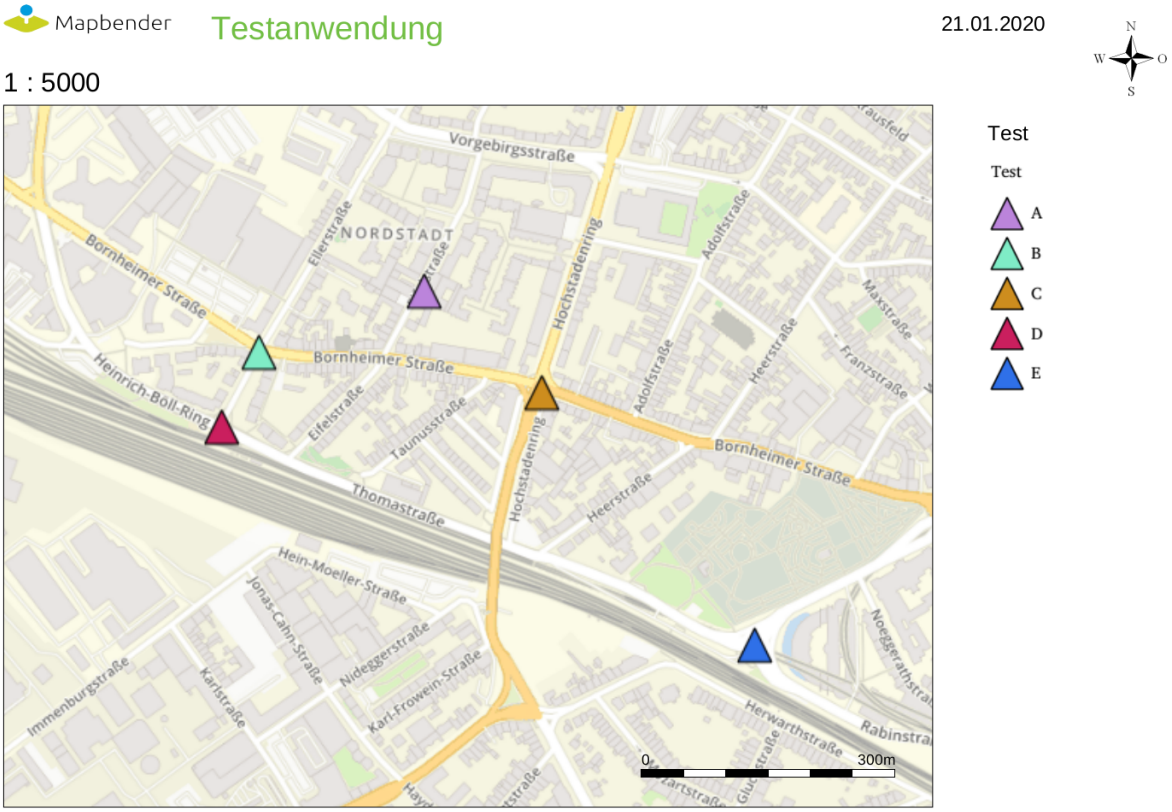
Farbige Texte

Der Text der Druckvorlage kann vielfältig angepasst werden. Neben der Schriftgröße besteht die Möglichkeit, die Farbe des Textes zu verändern. Hierfür wird ein Textfeld erzeugt **Menü: Einfügen -> Textfeld**. Soll der Text als dynamisches Element eingebunden werden, wird dieser auf der vorher festgelegten nicht druckbaren Ebene eingefügt und benannt. Hier wurde als Beispiel das dynamische Element “title” gewählt. Um den Text oder den Platzhalter zu färben, Text innerhalb des Textfeldes (hier: “title”) markieren. Die Änderung der Farbe kann rechts neben der Vorlage unterhalb des Reiters **Eigenschaften -> Zeichen** vorgenommen werden.



Alternativ kann über **Rechtsklick auf den markierten Text -> Zeichen ...** unter **Schrifteffekte -> Schriftfarbe** die Farbe verändert werden.

Die Veränderung der Farbe des dynamischen Feldes “title” in blau kann im Druck wie folgt aussehen:



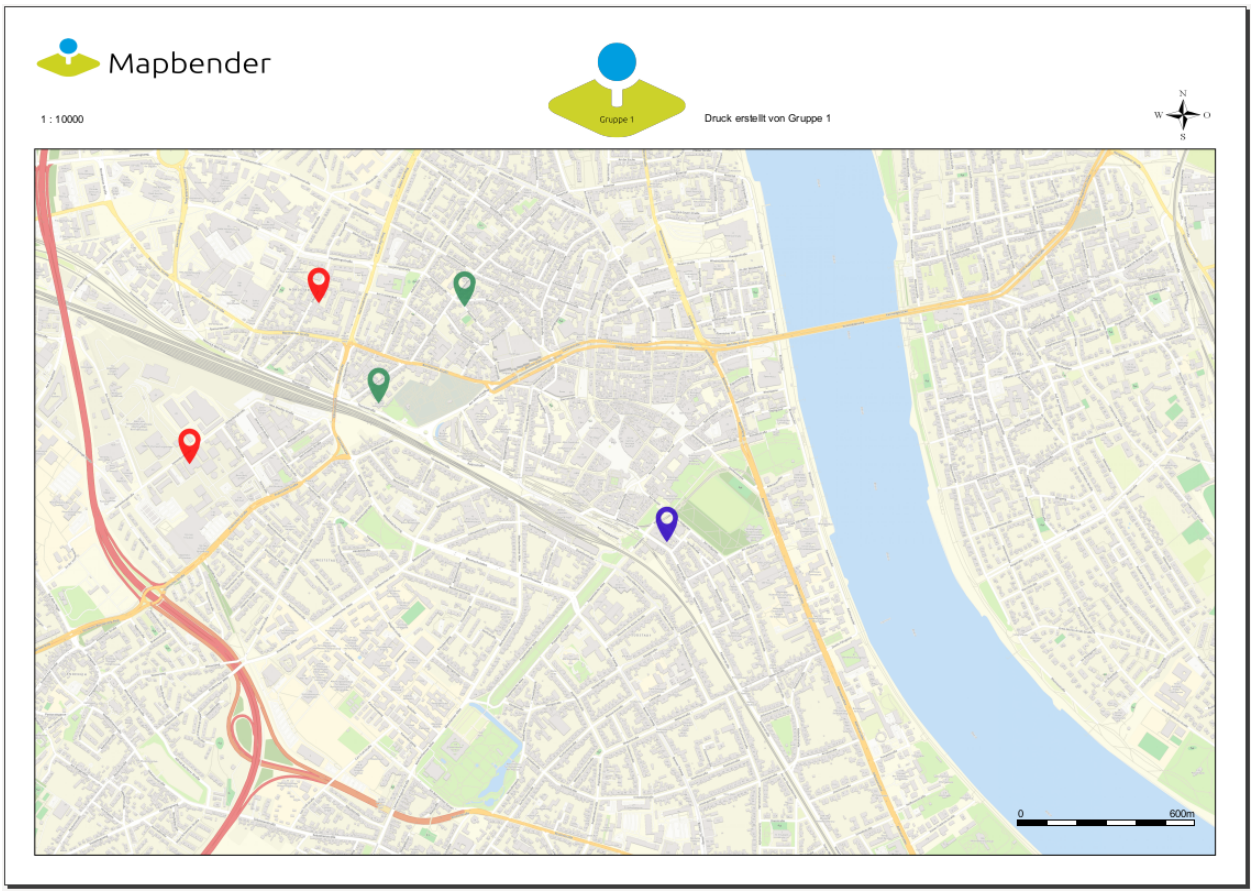
Analog zu der Veränderung der Schriftfarbe wird auch die Veränderung der Schriftgröße durchgeführt.

Dynamische Bilder und dynamische Texte

Gruppenabhängig können in der Druckausgabe unterschiedliche Bilder oder Beschreibungen (z.B. Logo und Bezeichnung der Gemeinde) ausgegeben werden. Hierzu gibt es die Platzhalter “dynamic_image” und “dynamic_text”. Beide Elemente können in der ODG-Druckvorlage in die nicht druckbare Ebene eingefügt, umbenannt (**Menü: Format -> Name... bzw. Kontextmenü des Elements -> Name...**) und entsprechend platziert werden.

Hinweis: Es kann immer nur ein Gruppenbild und eine Gruppenbeschreibung ausgegeben werden. Mapbender zeigt immer die zuerst aufgeführte Gruppe an. Ist der Benutzer in der Gruppe “intern” und “Gruppe 1”, dann wird “intern.png” als Bild genommen und die Gruppenbeschreibung der Gruppe “intern” als dynamischer Text ausgegeben.

Ein gruppenabhängiger Druck könnte bei einer Gruppe namens “Gruppe 1” wie folgt aussehen:



Zur Nutzung dieser Funktion müssen Gruppen mit Benutzern erstellt und den Anwendungen die jeweiligen Gruppen zugewiesen werden. Weitere Informationen zur Funktionsweise der Gruppen- und Benutzerverwaltung stehen im *Mapbender Schnellstart*.

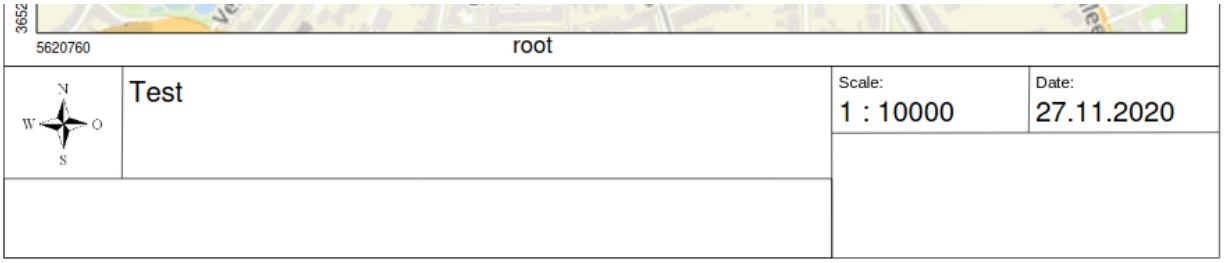
Dynamisches Bild

Sobald "dynamic_image" im Drucklayout vorliegt, wird nach einem Bild mit dem Namen der ersten zugewiesenen Gruppe gesucht und dieses im Bereich des Elements "dynamic_image" ausgegeben. Hierbei wird die Höhe zur Orientierung verwendet und die Breite entsprechend angepasst. Die verschiedenen Bilder je Gruppe werden im Ordner **config/MapbenderPrintBundle/images/** unter dem jeweiligen Namen der Gruppe abgelegt (z.B. Gruppenname ist "Gruppe 1", dann lautet der Name des Bildes Gruppe 1.png).

Dynamischer Text

Über das Element "dynamic_text" wird die Gruppenbeschreibung der ersten zugewiesenen Gruppe im Ausdruck eingetragen. Das Textfeld verhält sich genauso wie andere Textfelder und kann beliebig viele Zeichen enthalten. Sie können den dynamischen Text unabhängig von dem dynamischen Bild einbinden und beispielweise für Copyright-Hinweise nutzen.

Außerdem kann der Name des eingeloggten Nutzers ausgedruckt werden. Dafür muss der Platzhalter **user_name** an der gewünschten Stelle im Template eingefügt werden. Wenn kein Nutzer eingeloggt ist, dann bleibt beim Druck dieser Bereich frei.



Der Druckvorgang

Druck von Information für ein ausgewähltes Objekt

Es können Informationen zu einem ausgewählten Objekt ausgedruckt werden. Ein Objekt kann über die Digitalisierung (Digitizer) oder die Informationsabfrage (FeatureInfo) selektiert werden.

Der feature_type-name und die selektierte object-id wird an den Druck weitergeleitet. Dadurch erhält Mapbender alle Informationen, um zu den selektierten Objekten die Sachdaten zu ermitteln und in Feldern im Drucktemplate auszugeben. Im Drucktemplate wurde festgelegt, welche Daten ausgegeben werden sollen.

Im Folgenden wird beschrieben, wie dieses Verhalten konfiguriert werden kann. Die Dokumentation bezieht sich auf die poi-Tabelle, die im digitizer-Beispiel verwendet wird.

Sie finden die Konfiguration und ein Beispiel-Drucktemplate im Workshop/DemoBundle unter https://github.com/mapbender/mapbender-workshop

Die folgenden Schritte müssen durchgeführt werden:

1. Erzeugen eines Drucktemplates, das auf die Objektspalten verweist
2. Definition eines featureTypes und Verweis auf das neue Drucktemplate in der doctrine.yaml
3. Druck über die Informationsabfrage aufrufen
4. Alternativ: Aufruf des Drucks über die Digitalisierung

1. Erzeugen einer Druckvorlage, die auf die Objektspalten verweist

Im Drucktemplate ein Textfeld für die Informationen definieren, die für das selektierte Objekt ausdruckt werden sollen. Der Textfeldname hat immer den Prefix *feature.* gefolgt vom Namen der Spalte.

```
feature.name for column name of table poi
```

2. Definition eines featureTypes und Verweis auf das neue Drucktemplate in der doctrine.yaml

```
parameters:
  featureTypes:
    feature_demo:
      connection: search_db   # Name der Datenbankverbindung von der doctrine.yaml
      table: public.poi       # Tabellenname, in der sich die Objekte befinden
      uniqueId: a_gid         # Spaltennname mit der eindeutigen ID
      geomType: point         # Geometrietyp
```

(continues on next page)

(continued from previous page)

```
      geomField: geom          # Spaltenname, in der die Geometrie gespeichert ist
      srid: 4326               # EPSG-Code der Daten
      print:                   # Drucktemplate für den Druck selektierter Objekte
        templates:
         - template: a4_portrait_official_feature_data_demo
           label: Demo with feature information print (portrait)
         - template: a4_landscape_official_feature_data_demo
           label: Demo with feature information print (landscape)
```

3. Aufruf des Drucks über die Informationsabfrage

Bemerkung: Die Informationsabfrage (FeatureInfo) ist die Ausgabe von Informationen von einem OGC WMS Service. Sie gibt Informationen zu Objekten an einer Klickposition aus.

Beim Konfigurieren eines WMS, muss ein Link mit der folgenden Referenz generiert werden, die den Druck mit Objektinformationen anstößt.

Der folgende Code ist ein Beispiel für ein MapServer FeatureInfo-Template.

```
<table>
<script src="http://code.jquery.com/jquery-latest.js"></script>
<tr>
<td class="th_quer">Drucken</td>
<td><a href="" onclick="parent.$('.mb-element-map').data('mapQuery').olMap.setCenter([[x],[y]]);parent.
↪$('.mb-element-printclient:parent').data('mapbenderMbPrintClient').printDigitizerFeature('feature_demo
↪',[gid]);parent.$('.mb-element-featureinfo:parent').data('mapbenderMbFeatureInfo').deactivate();
↪return false">print feature information</a>
</td>
</tr>
</table>
```

Die Informationsabfrage (FeatureInfo) öffnet einen Dialog mit dem Link *print feature information*. Mit Klick auf den Link, öffnet sich ein Druckdialog, der das Drucktemplate für das selektierte Objekt anbietet.

Das gewünschte Gebiet kann auswählt und ein PDF erzeugt werden. Das PDF beinhaltet die Informationen für das selektierte Objekt.

4. Alternativ: Aufruf des Drucks über die Digitalisierung

Die Funktion kann auch in die Digitalisierung eingebunden werden. Im Digitalisierungsdialog wird dann ein neuer Button *Drucken* angeboten.

Zum Aktivieren der Funktion müssen die folgenden Parameter zur Digitalisierungskonfiguration hinzugefügt werden.

```
printable: true
```

Mit Klick auf den Druckbutton, öffnet sich ein Druckdialog, der das definierte Drucktemplate für das selektierte Objekt zur Verfügung stellt.

Das gewünschte Gebiet kann auswählt werden und ein PDF erzeugt. Das PDF beinhaltet die Informationen für das selektierte Objekt.

Note: Die Flexibilität, den Druckrahmen zu verschieben, hindert den Anwender nicht daran, den Rahmen in einen Bereich zu verschieben, der nicht das ausgewählte Objekt enthält. Die ausgedruckte Objektinformation passt dann nicht zur Darstellung in der Karte.

Warteschleifendruck

Der Warteschleifendruck ist ein Druckfeature in Mapbender, welches einen erweiterten Hintergrunddruck erlaubt. Dieses experimentelle Feature ist seit Mapbender 3.0.8 implementiert. Es ist standardmäßig nicht aktiviert, da bei komplexeren Systemen Probleme mit der Cache-Speicher-Regeneration auftreten können. Sobald aktiviert, kann das Feature entweder händisch über die Kommandozeile angestoßen oder über einen Cronjob automatisiert werden. Der Warteschleifendruck hilft dabei, ressourcenintensive Druckjobs mit langen Ausführungszeiten zu verbessern, indem diese in eine Warteschleife, die im Hintergrund abgearbeitet wird, ausgelagert werden. Währenddessen können Sie mit Mapbender anderweitig weiterarbeiten.

Warteschleifendruck: Konfiguration

Um den Warteschleifendruck zu aktivieren, muss die parameters.yaml-Datei wie folgt ergänzt werden:

```
mapbender.print.queueable: true
```

Dabei muss “Modus” auf die Option “Warteschleife” gesetzt werden, da sonst standardmäßig [der] Direktdruck ([“Modus”] “direkt”) eingestellt ist.

Warteschleifendruck: Kommandozeilenbefehle

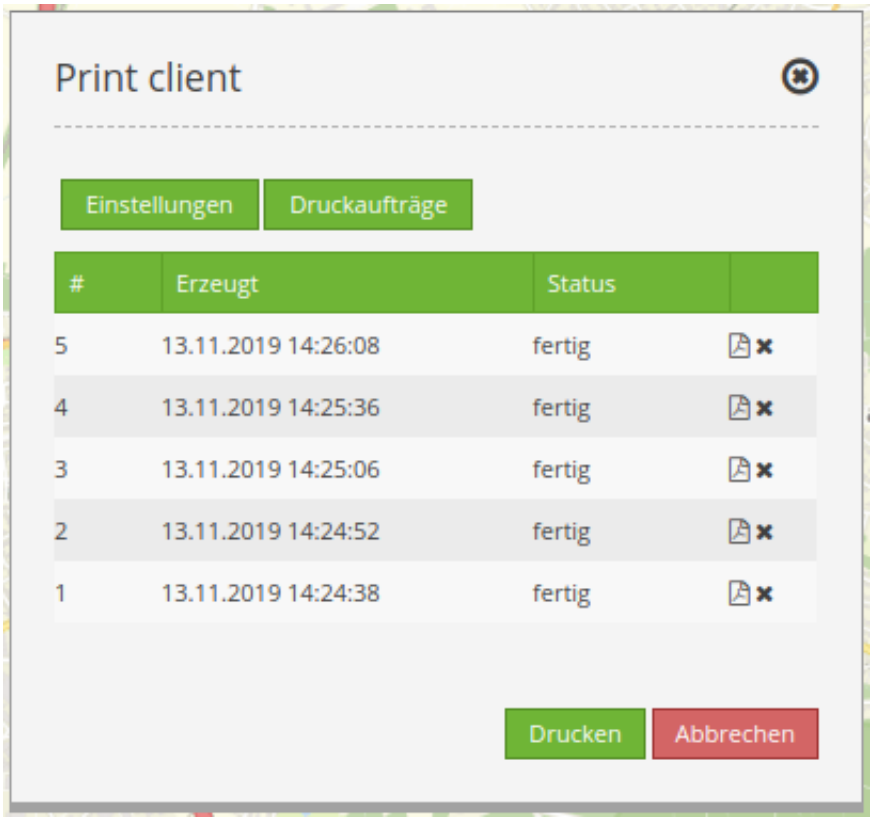
Nach Initialisierung des Warteschleifendrucks stehen die folgenden Funktionen über die Kommandozeile zur Ausführung des Drucks zur Verfügung[:]

```
mapbender:print:queue:clean
mapbender:print:queue:dumpjob
mapbender:print:queue:gcfiles
mapbender:print:queue:next
mapbender:print:queue:repair
mapbender:print:queue:rerun
mapbender:print:runJob
```

Hint: Bemerkung: Zur Ausführung der Befehle muss sich der Benutzer im application-Verzeichnis befinden und bin/console den jeweiligen Befehlen voranstellen. Zur genauen Vorgehensweise siehe die Informationen auf der Seite *bin/console Befehle*.

Warteschleifendruck: Durchführung

Der Tab „Einstellungen“ bietet die vom Direktdruck gewohnten Druckoptionen. Nachdem der Warteschleifendruck eingerichtet wurde, kann neben dem Tab „Einstellungen“ über einen Button die neu erscheinende Funktion „Druckaufträge“ angewählt werden. Hier finden sich chronologisch alle Druckaufträge aufgelistet, die der User über das Mapbender-Interface wie gewohnt erstellt.



Neu ist, dass die Druckaufträge im Hintergrund erst nach Start des Prozesses

```
bin/console mapbender:print:queue:next --max-jobs=0 --max-time=0
```

in der Kommandozeile ausgeführt werden. Er bewirkt, dass nach dem Klick auf den Drucken-Button der Druck entsprechend seiner Position in der Warteliste über die Kommandozeile durchführt wird. Der Prozess wird über die Tastenkombination “Strg+C” beendet. Nach abgeschlossener Generierung der Dateien (erkennbar über den Status “fertig” in der Liste der Druckaufträge) können diese über den PDF-Button geöffnet werden können.

Speicherbegrenzungen

Warteschleifendruck

Da der Warteschleifendruck möglicherweise speicherintensiver sein kann als anfangs in Ihren PHP-Einstellungen festgelegt, kann der benötigte Speicher durch manuelle Konfiguration erhöht werden. Dies ist für Anwender, die mit größeren Ausdrucken arbeiten möchten, besonders von Vorteil. Der Parameter *mapbender.print.queue.memory_limit* (string; Standard: 1G) muss angepasst werden, um die Speicherbegrenzung speziell für den Warteschleifendruck zu erhöhen. Vorsicht: Dieser Parameter erlaubt keine “null”-Werte.

Note: Bemerkung: Reduzieren Sie die Speicherbegrenzung nicht.

Direktdruck

Über den Parameter *mapbender.print.memory_limit* (string or null; Standard: null) kann das Speicherlimit auch für den Direktdruck angepasst werden (mögliche Werte sind z.B. 512M, 2G, 2048M, etc.). Ist der Parameter "null" eingestellt, passt sich der Druck an die vorgegebene php.ini-Begrenzung an, der Wert "-1" steht für unbegrenzte Speichernutzung.

WMS-Kachelgröße begrenzen

Sofern der Druck einen WMS-Dienst nicht erfolgreich in die PDF-Datei exportieren sollte, muss in der parameters.yaml-Datei eine Ergänzung vorgenommen werden. Das hat damit zu tun, dass unter bestimmten Umständen die angeforderte Pixelausdehnung für den WMS zu groß ist, sodass der Dienst keine Bilder mehr liefert.

```
mapbender.imaageexport.renderer.wms.max_getmap_size: 8192
```

Durch diese Begrenzung werden die größtmöglichen WIDTH=- und HEIGHT=-Werte für die Exportanfrage festgelegt. Im GetCapabilities-Request des jeweiligen Dienstes wird die maximale Auflösung unter `MaxWidth` bzw. `MaxHeight` definiert, sodass der getCapabilities-Request das Limit bereits vorgibt - bei *8192* handelt es sich um den Standardwert, der eventuell weiter angepasst werden muss. Die oben genannten Parameter können auch unabhängig voneinander definiert werden. Verwenden Sie `mapbender.imaageexport.renderer.wms.max_getmap_size.x` für den **WIDTH=**- und `mapbender.imaageexport.renderer.wms.max_getmap_size.y` für den **HEIGHT=**-Parameter.

3.4 Share-Elemente

3.4.1 Share-Elemente: Überblick

Mapbender bietet eine Vielzahl an Elementen, die die gemeinsame Arbeit mit Kartenanwendungen erleichtern:

- *URL teilen (Share URL)* ermöglicht das schnelle Teilen von selbstkonfigurierten Kartenanwendungszuständen via URL,
- *Ansichtsverwaltung (View Manager)* speichert Kartenzustände und erstellt eine Liste dieser in der Sidepane,
- *Anwendung wechseln (Application Switcher)* ermöglicht den einfachen Wechsel zwischen Anwendungen,
- *Persistente Kartenzustände* ermöglicht das Speichern des Kartenzustands im Browser.

Außerdem übergibt Mapbender einige Parameter automatisch in der Anwendungs-URL. Dadurch ist es möglich, aktuell gewählte Einstellungen durch Teilen der URL an andere weiterzuleiten.

Die URL beinhaltet folgende Parameter:

- Kartenposition
- Maßstab

- Drehung
- Räumliches Referenzsystem

Diese Funktion muss nicht extra konfiguriert werden. Sie ist immer aktiviert.

Wenn eine URL in einem neuen Browserfenster geöffnet wird, dann erfolgt auch die Übernahme der genannten Kartenparameter. Änderungen können über die Browsernavigation vor- bzw. zurückgenommen ("Weiter" oder "Zurück") werden.

Nach dem Neuladen des Browserfensters werden Anwender zum jeweiligen Kartenausschnitt zurückgesetzt, sofern die Parameter nicht in der URL übergeben werden. Soll zur Start-Konfiguration zurück navigiert werden, dann muss die Anwendung entweder neu geöffnet werden oder eine manuelle Anpassung der URL erfolgen.

Note: Folgende Eigenschaften werden nicht in der URL hinterlegt: temporäre Layer-Auswahl und Layer-Sortierung, temporäre Geometrien, angepasste Transparenzwerte, interaktiv hinzugefügte Instanzen oder andere Laufzeitergänzungen.

3.4.2 URL teilen (Share URL)

Das Teilen von URLs kann durch den Einbau eines entsprechenden Elements vereinfacht werden. Dieses wird als Button in der oberen Werkzeugleiste oder der Fußzeile eingebunden.

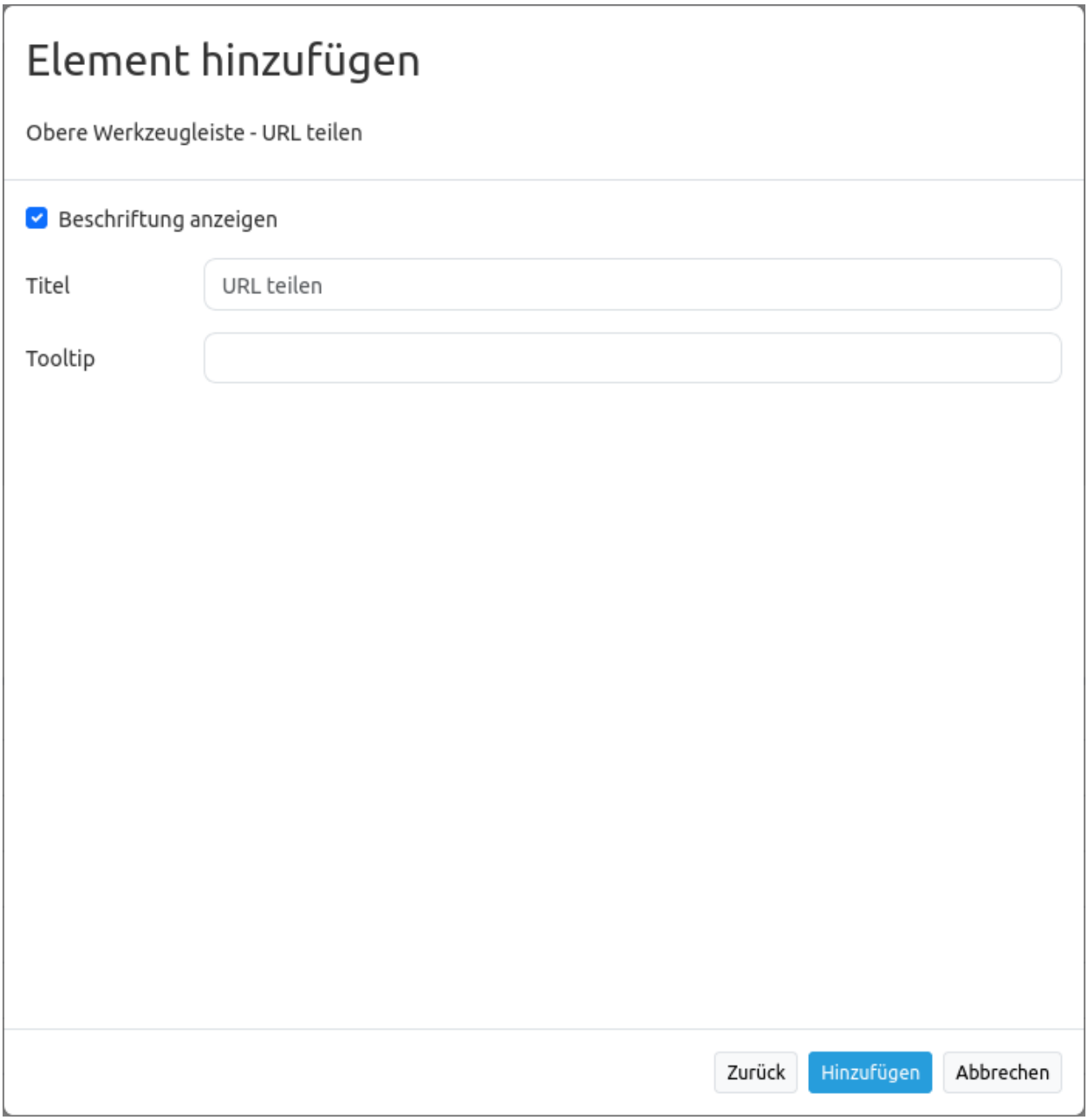
Durch einen Klick auf das Element wird die URL in der Zwischenablage gespeichert. Gängige Browser-Interaktionen (z.B. In neuem Tab öffnen) sind ebenfalls möglich.

Das Element speichert folgende Informationen:

- Grundlegende Kartenparameter (Kartenposition, Maßstab, Drehung, Referenzsystem)
- Layer- und Layerset-Einstellungen (Auswahl sowie Transparenz)

Note: Hinweis: Folgende Informationen werden nicht übergeben: interaktiv hinzugefügte Instanzen (WMS laden), interaktiv entfernte Instanzen (Ebenenbaum Kontextmenü) sowie Änderungen der Layerreihenfolge (Drag & Drop).

Konfiguration



- **Beschriftung anzeigen:** Schaltet die Beschriftung des Buttons an/aus (Standard: an).
- **Titel:** Titel des Elements. Entspricht dem Button-Titel.
- **Tooltip:** Text, der angezeigt wird, wenn der Mauszeiger längere Zeit über dem Element verweilt.

YAML-Definition

Diese Vorlage kann genutzt werden, um das Element in einer YAML-Anwendung einzubinden.

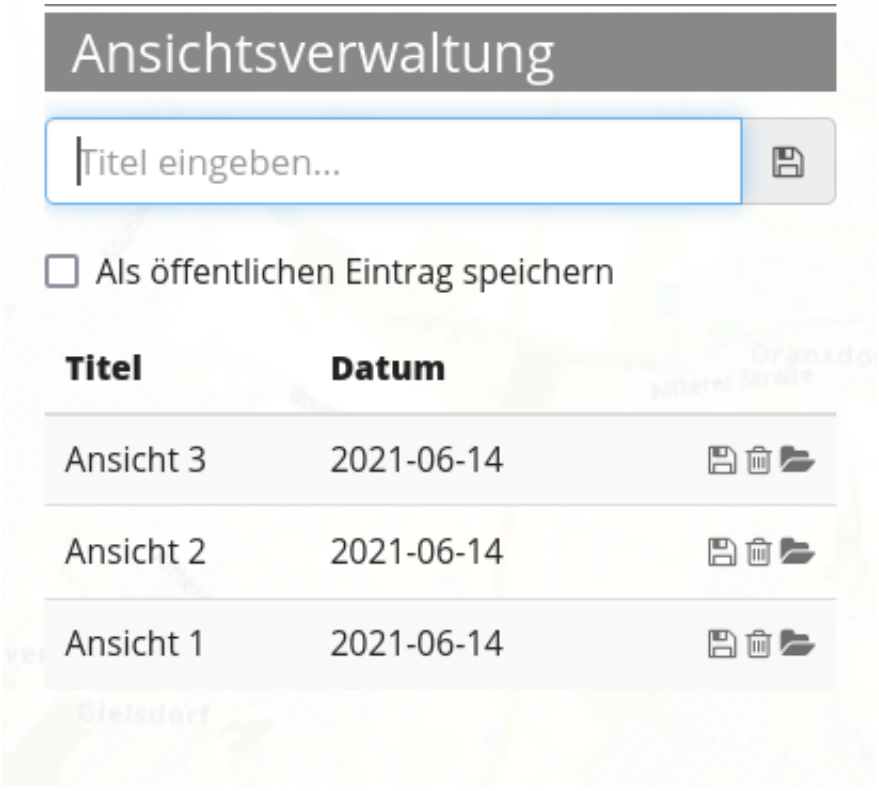
```
title: Teile diese Kartenansicht          # Optionaler Titel, standardmäßig wird "URL teilen" verwendet
↪(string oder leer).
class: Mapbender\CoreBundle\Element\ShareUrl
tooltip: Ich werde beim Hovern angezeigt  # Optionaler Titel, entpricht bei Weglassung dem Titel
↪(string oder leer).
label: true                               # Aktiviert Darstellung des Titels, zeigt bei FALSE das
↪Element Icon an (Standard: true).
```

3.4.3 Ansichtsverwaltung (View Manager)

Das Element Ansichtsverwaltung erlaubt das Speichern und Wiederverwenden von Kartenansichten. Folgende Kartenparameter sind Teil einer Ansicht:

- Drehung
- Kartenposition
- Koordinatenreferenzsystem
- Layer aus *Datenquellen (Sources)*
- Ausgewählte *Layersets*
- Maßstab
- Transparenz

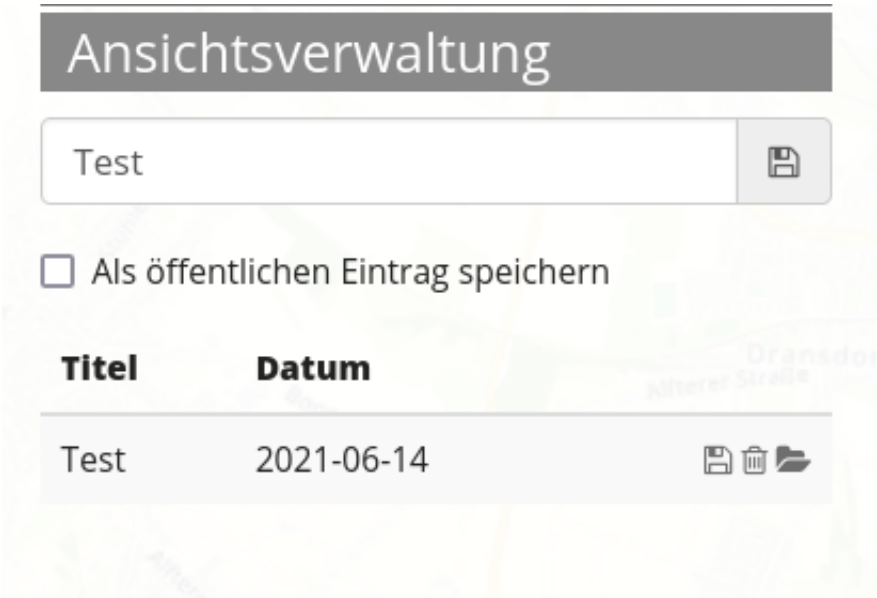
Gesetzte Kartenansichten bleiben nach Neuladen der Anwendung erhalten.



Note: **Hinweis:** Die Ansichtsverwaltung wird bisher nur in der Sidepane unterstützt.

Nutzung

Um eine neue Kartenansicht zu speichern, muss zunächst ein Titel für deren Re-Identifikation vergeben werden. Danach kann die Kartenansicht durch einen Klick auf den Speichern-Button zur Liste hinzugefügt werden.



In seiner einfachsten Form kann das Element zur Wiederverwendung von Kartenansichten verwendet werden. Diese Option ist immer vorhanden: Die zuvor gespeicherte Kartenansicht wird nach Klick auf den Abrufen-Button wiederhergestellt. Gespeicherte Ansichten können außerdem überschrieben oder gelöscht werden. Die Einträge in der Sidepane werden dabei entsprechend aktualisiert.

Die folgenden Konfigurationen werden **nicht** in die Ansichtsverwaltung integriert:

- Interaktiv hinzugefügte Instanzen (*WMS laden (WMS Loader)*)
- Interaktiv entfernte Instanzen (*Ebenenbaum (Layertree)* Kontextmenü)
- Alle Werte für WMS-Dimensionen
- Dynamisch veränderte Geometrien (*Digitizer* etc.)

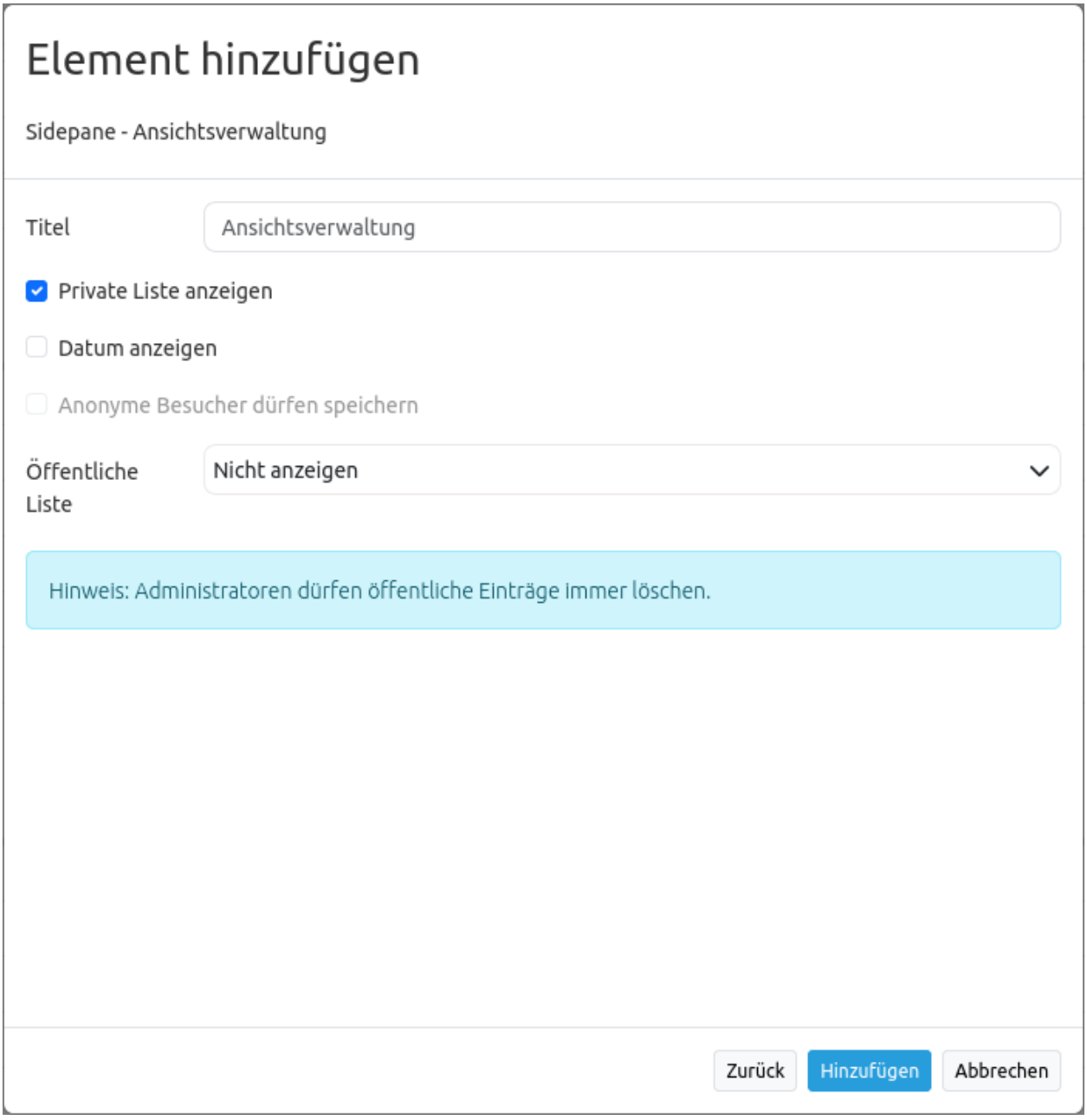
Zugriffsrechte

Jede Kartenansicht ist einer Anwendung zugeordnet und wird weiter in öffentlich und privat unterteilt. Die Elementkonfiguration enthält die Rechteeinstellungen zum Lesen, Speichern und Löschen von Kartenansichten in öffentlichen Listen. Weiterhin kann hier über Checkboxen definiert werden, ob private Listen die Anzeige privater Kartenansichten ermöglichen sollen, der beim Speichern erstellte Zeitstempel in der Liste angezeigt wird und ob anonyme Besucher öffentliche Kartenansichten speichern dürfen.

Zugriffsrechte müssen für den root Nutzer nicht für öffentliche Ansichten gesetzt werden. Der Administrator kann diese automatisch erzeugen, überschreiben oder löschen.

Anonyme Nutzer sind generell von der Arbeit mit privaten Ansichten ausgeschlossen. Sie können öffentliche Einträge außerdem niemals löschen. Ihre Zugriffsrechte für das Speichern sowie Überschreiben von öffentlichen Ansichten wird über die Checkbox "Anonyme Besucher dürfen speichern" gesteuert. Bei Deaktivierung der Option verbleibt anonymen Benutzern noch ein Lesezugriff. Sollen Anonyme Nutzer komplett ausgeschlossen werden, dann kann eine ROLE_USER Zugriffsbeschränkung für das gesamte Element gesetzt werden.

Konfiguration



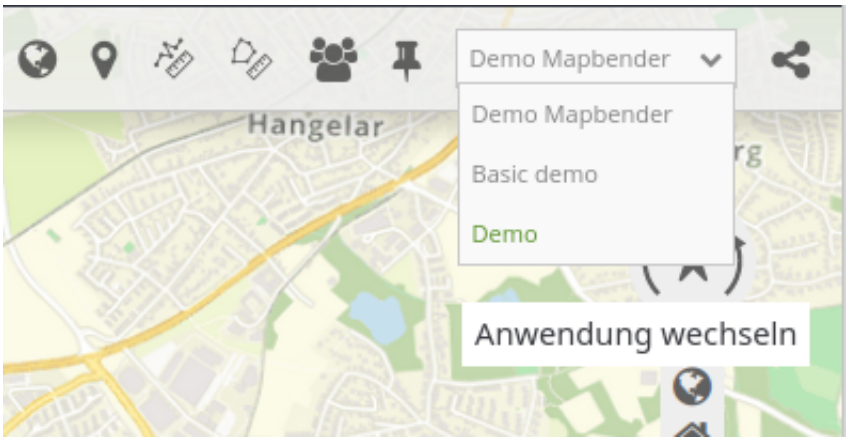
YAML-Definition

Diese Vorlage kann genutzt werden, um das Element in einer YAML-Anwendung einzubinden.

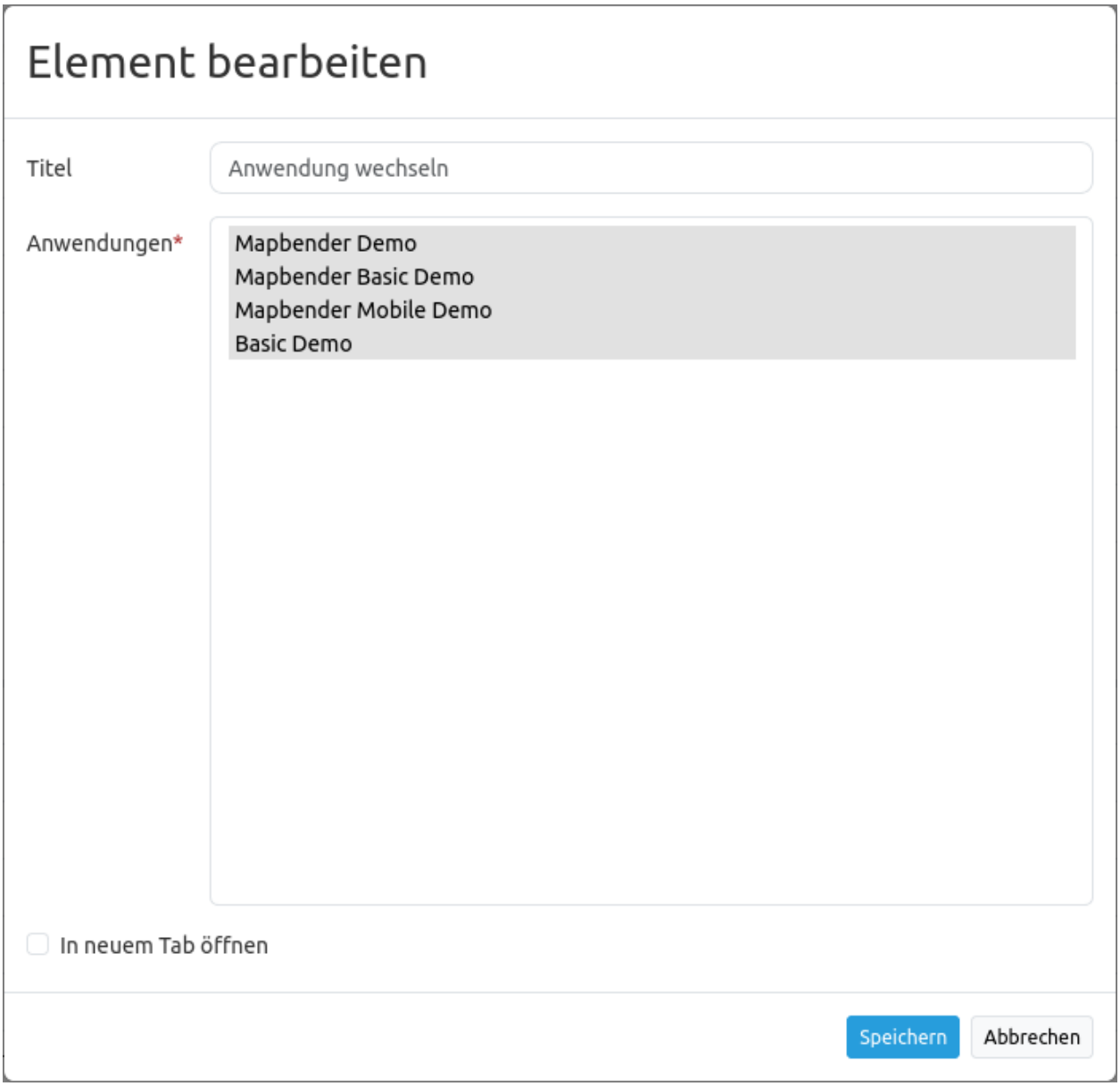
```
viewmanager:
  title: View Manager                                  # Titel des Elements.
  class: Mapbender\CoreBundle\Element\ViewManager      # Klasse des Elements.
  publicEntries: rw                                    # String oder leer (Falsche Werte deaktivieren
↪öffentliche Ansichten komplett); andere erlaubte Werte sind ro (nur Lesezugriff), rw (Lese- und
↪Schreibzugriff), rwd (Lese- und Schreibzugriff sowie Löscherlaubnis) (Standard: ro).
  privateEntries: true                                 # Schaltet Privatnutzerzustände an mit vollem
↪Zugriff auf Optionen "Speichern", "Wiederverwenden" sowie "Löschen" (Standard: true).
  allowAnonymousSave: true                             # Gibt Speicherrecht für öffentliche Ansichten
↪auch an anonyme Nutzer (Standard: false).
  showDate: true                                       # Zeigt Datum der Erzeugung bzw. Aktualisierung
↪(Standard: true)
```

3.4.4 Anwendung wechseln (Application Switcher)

Dieses Element stellt ein Auswahlbox bereit, über die zwischen Anwendungen gewechselt werden kann. Dabei bleibt der aktuelle Kartenausschnitt erhalten.



Konfiguration



- **Title:** Titel des Elements. Dieser wird angezeigt, wenn der Mauszeiger eine längere Zeit über der Auswahl verweilt.
- **Anwendungen:** Wählen Sie die Anwendungen, die zur Auswahl erscheinen sollen.
- **In neuem Tab öffnen:** Definiert, ob die Anwendung beim Wechsel in einem neuen Browser-Tab geöffnet werden soll.

YAML-Definition

Diese Vorlage kann genutzt werden, um das Element in einer YAML-Anwendung einzubinden.

```
title: Choose an Application              # Text wird als Tooltip angezeigt
class: Mapbender\CoreBundle\Element\ApplicationSwitcher
applications: ["mapbender_user","mapbender_user_basic"]     # Anwendungen, die zur Auswahl stehen sollen
open_in_new_tab: true   # false/true Anwendung in neuem Browser-Tab öffnen
```

3.4.5 Persistente Kartenzustände

Diese Funktion macht bestimmte Kartenparameter und -einstellungen persistent: Eine Kartenanwendung kann dann in einem neuen Browser-Tab geöffnet werden, ohne dass dabei die Kartenparameter auf die Standardeinstellung zurückgesetzt werden.

Persistente und damit wiederherstellbare Informationen umfassen:

- Kartenposition, Maßstab, Koordinatenreferenzsystem, Drehung
- Aktivierte/deaktivierte Layer
- Aktivierte/deaktivierte Layersets
- Transparenz

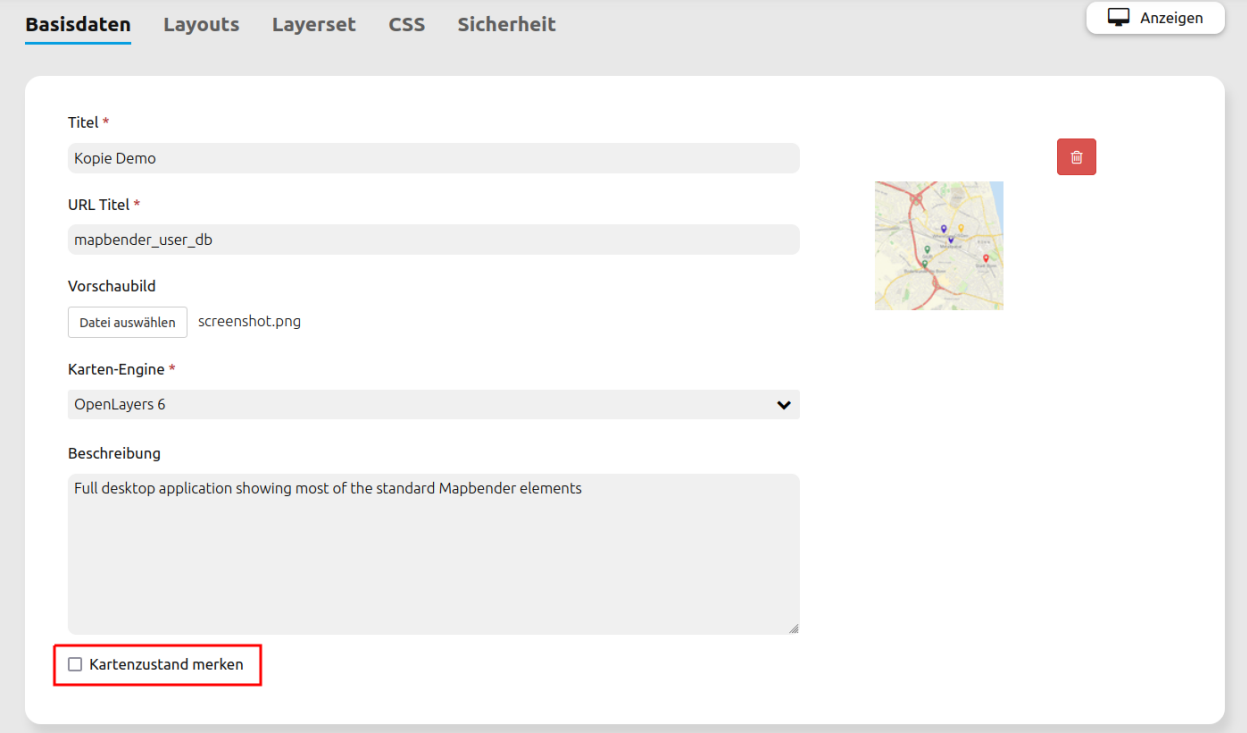
Note: Hinweis: Die Persistenz beruht nur auf dem eigenen Browserspeicher und ist damit rein lokal. Dies trifft auch auf Systeme mit mehreren Anwendern zu. Die Funktion hat außerdem keinerlei Auswirkung auf den Mapbender Login.

Folgende Kartenparameter und Einstellungen sind nicht Teil persister Kartenzustände:

- WMS-Dimensionen
- interaktiv hinzugefügte Instanzen (WMS laden)
- interaktiv entfernte Instanzen (Ebenenbaum Kontextmenü)
- Umstrukturierung von Instanzen im Ebenenbaum (Drag & Drop)
- FeatureInfo Konfiguration für Instanz(en)

Konfiguration

Persistente Kartenzustände werden über eine Checkbox für jede Anwendung unter dem Tab “Basisdaten” einzeln aktiviert.



Die Einbettung erzeugt eine neue Spalte in der Tabelle *mb_core_application*. Es muss deshalb eine Datenbankaktualisierung mit *bin/console doctrine:schema:update –force* durchgeführt werden.

YAML-Definition

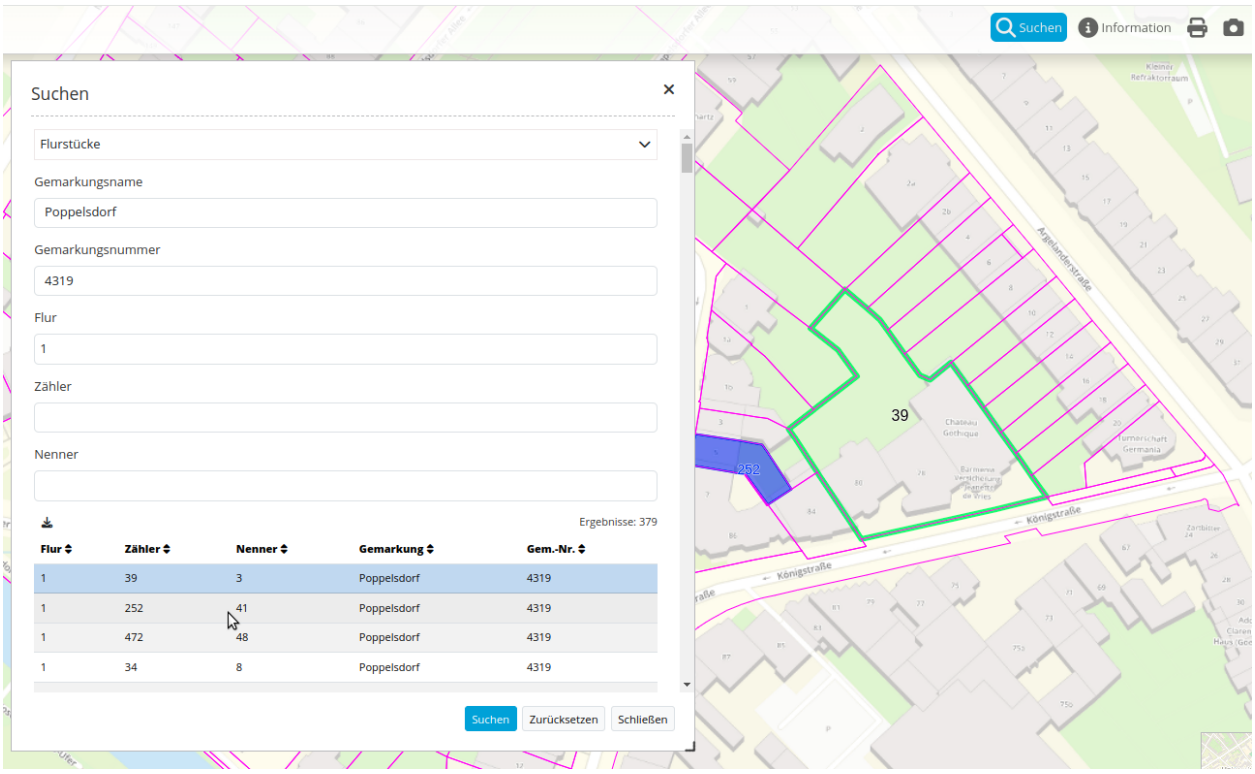
Diese Vorlage kann genutzt werden, um das Element in einer YAML-Anwendung einzubinden. Zu beachten ist hierbei der neue Parameter *persistentView*. Wird dieser weggelassen, dann entspricht dies *false*.

```
parameters:
    applications:
        mapbender_user:
            title: Mapbender Demo Map
            screenshot: screenshot.png
            published: true
            persistentView: true      # Parameter zur Aktivierung der persistenten Kartenzustände
            template:  Mapbender\CoreBundle\Template\Fullscreen
```

3.5 Suchen

3.5.1 Suchen (Search Router)

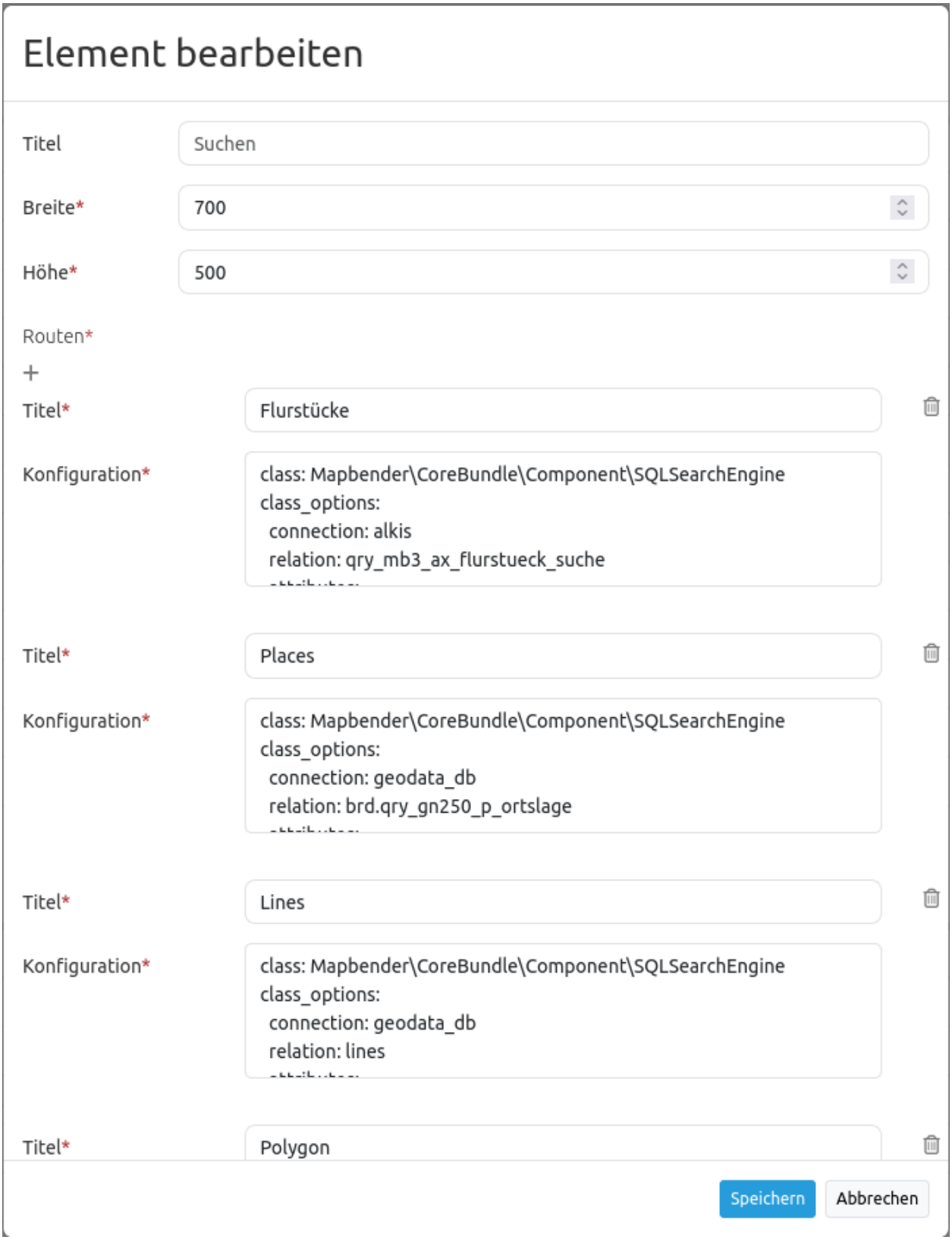
Dieses Element erzeugt ein Suchformular mit Trefferausgabe, über das Objekte innerhalb der Karte lokalisiert werden können. Sowohl eine Verbindung zur Datenbank als auch das Formular und die Trefferausgabe sind über das *Backend* konfigurierbar. Derzeit werden PostgreSQL-Tabellen als Grundlage für Suchen unterstützt.



Konfiguration

Note: **Hinweis:** Die Suche greift auf Tabellen in einer Datenbank zu. Dafür muss die Datenbank in Mapbender bekannt gegeben werden. Informationen dazu finden sich unter *YAML Konfiguration (Konfigurations- und Anwendungsdateien)*.

Das Element kann entweder in der Sidepane oder als Button in der oberen Werkzeugleiste integriert werden. Zu der Konfiguration des Buttons besuchen Sie die Dokumentationsseite unter *Button*.



- **Titel:** Titel des Elements. Dieser wird unter der *Layouts*-Liste sowie als Titel in der Seitenleiste oder als Titel des Suchdialogs (via Button) angezeigt.

- **Breite:** Breite des Dialog-Fensters (im Kartenbereich).
- **Höhe:** Höhe des Dialogs-Fensters (im Kartenbereich).

Über + bei *Routen* können mehrere Suchen erstellt werden. Jede Suche beinhaltet die Felder *Titel* und *Konfiguration*. Der eingegebene Titel ist in der Suche über eine Auswahlbox selektierbar. So können mehrere Suchen unterschieden und ausgewählt werden.

- **Routen:** Erstellung einer neuen Suche.
- **Titel**: Titel der Suche (erscheint erst, wenn über + bei *Routen* eine Suche hinzugefügt wurde).
- **Konfiguration**: Feld für die Konfiguration der Suche (erscheint erst, wenn über + bei *Routen* eine Suche hinzugefügt wurde).

Konfiguration der einzelnen Suchen

Die Definition der einzelnen Suchen erfolgt im YAML-Syntax im Textfeld *Konfiguration*.

Hier erfolgt die Definition:

- der Datenbankverbindung,
- der Suchtabelle bzw. Abfrage,
- des Formularaufbaus,
- der Aufbau der Trefferausgabe,
- sowie das Styling der Treffer in der Karte.

```
class: Mapbender\CoreBundle\Component\SQLSearchEngine
class_options:
    connection: geodata_db   # database alias from doctrine.yaml
    relation: polygons       # search table
    attributes:              # columns that are used in the form and in the result table
      - gid
      - name
      - type
      - city
    geometry_attribute: geom # reference to the geometry column
form:
    name:
        type: Symfony\Component\Form\Extension\Core\Type\TextType
        options:
            required: true
        compare: exact
    city:
        type: Symfony\Component\Form\Extension\Core\Type\TextType
        options:
            required: false
            label: Ort/Stadt
        compare: ilike
results:
```

(continues on next page)

(continued from previous page)

```
    view: table
    count: true
    exportcsv: true
    sortBy: name
    sortOrder: asc #desc
    zoomToResultExtent: true
    headers:
        gid: ID
        name: Name
        city: City/Town
    callback:
        event: click
        options:
            buffer: 10
            minScale: null
            maxScale: null
```

Class und Class Optionen

Die Konfiguration beginnt mit der Angabe der **class** (immer `class: Mapbender\CoreBundle\Component\SQLSearchEngine`) sowie Informationen zur Datenbankverbindung, der Suchtabelle, den verwendeten Spalten und der Angabe der Geometriespalte.

Formular

Im Bereich **form** können Textfelder und Auswahlboxen für das Suchformular definiert werden. Die jeweilige Tabellenspalte wird angegeben und mit Definitionen für *type*, *options* und *compare* versehen.

Typ

Das Formular unterstützt zwei Typen: *text* für Textfelder und *choice* für Auswahlfelder.

Für jede Tabellenspalte, die im Formular erscheinen soll muss eine Konfiguration erstellt werden. Die Konfiguration beginnt mit dem Spaltennamen (in den Beispielen sind es die Spalten *name* und *usertype*).

- **label** - Sie können den Parameter label für die Spaltenbeschriftung angeben. Fehlt der Parameter wird der Tabellenspalten-Name mit führendem Großbuchstaben verwendet.
- **required** - Sie können definieren, ob ein Suchfeld zwingend gefüllt sein muss (Standard: false). Mit `required: true` wird ein Pflichtfeld erzeugt. Die Suche kann dann nicht abgeschickt werden, bevor alle Pflichtfelder gefüllt sind. Überschriften der Pflichtfelder werden mit einem roten * angezeigt.

Sie können außerdem den Vergleichsoperator definieren. Siehe *Vergleichsmodus*.

Typ text

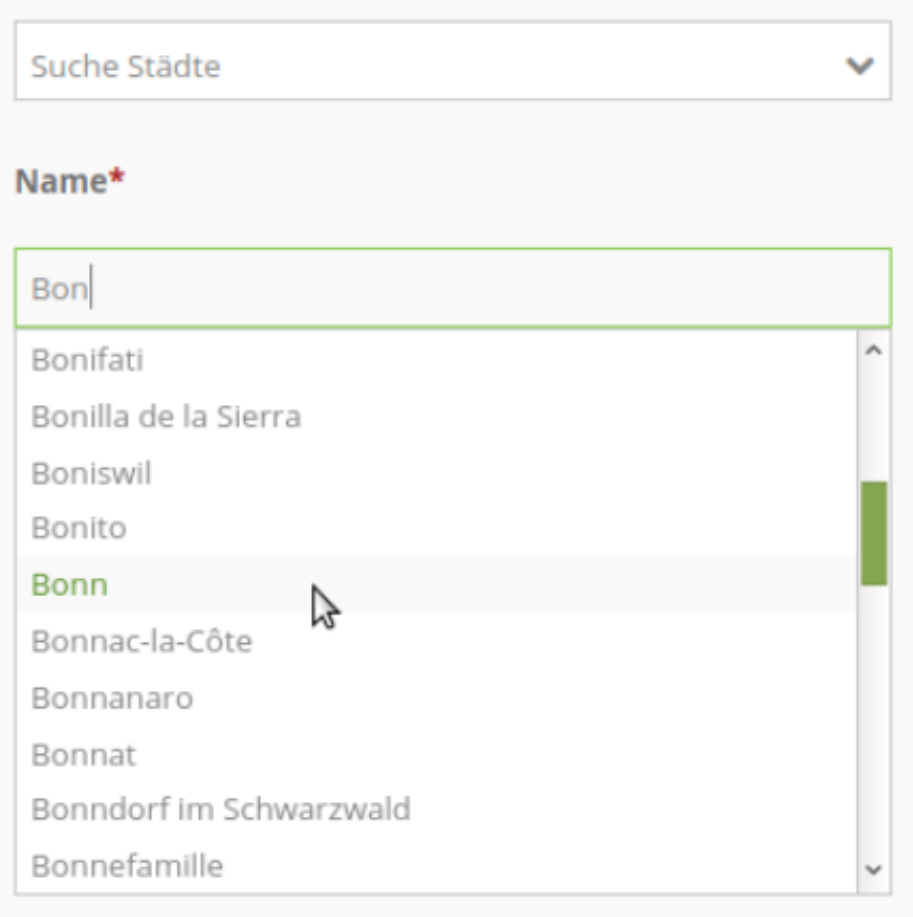
Über den Typ **text** können Textfelder in das Suchformular eingefügt werden.

Text unterstützt Autovervollständigung. Sofern sie diese nutzen möchten, müssen Sie zusätzliche Parameter hinzufügen.

Unterstützte autocomplete-Parameter:

- **data-autocomplete**: 'on' - Parameter zum Aktivieren der Autovervollständigung.
- **data-autocomplete-distinct**: 'on' - Gibt doppelte Ergebnisse der Autovervollständigung nur einmal aus.
- **data-autocomplete-using**: column1, column2 - Definiert weitere Spalte/n des Suchformulars, die bei der Autovervollständigung berücksichtigt werden sollen.

type text-Beispiel mit Autocomplete und Pflichtfeld:




```
form:
    name:
        type: Symfony\Component\Form\Extension\Core\Type\TextType
↪               # input box for text
        options:
            label: Name
            required: true
            attr:
```

(continues on next page)

(continued from previous page)

```
                data-autocomplete: 'on'           # aktiviere Autovervollständigung
                data-autocomplete-distinct: 'on'  # gleiche Treffer nur einmal anzeigen
                data-autocomplete-using: type, city   # Liste von weiteren Spalten für die
↪Autovervollständigung (kommasepariert)
        compare: exact
```

Typ choice

Der Typ **choice** ermöglicht die Definition von Auswahlboxen in dem Suchformular.

- **placeholder** - Es kann ein Platzhaltertext definiert werden. Dieser erscheint wenn noch keine Auswahl erfolgte.

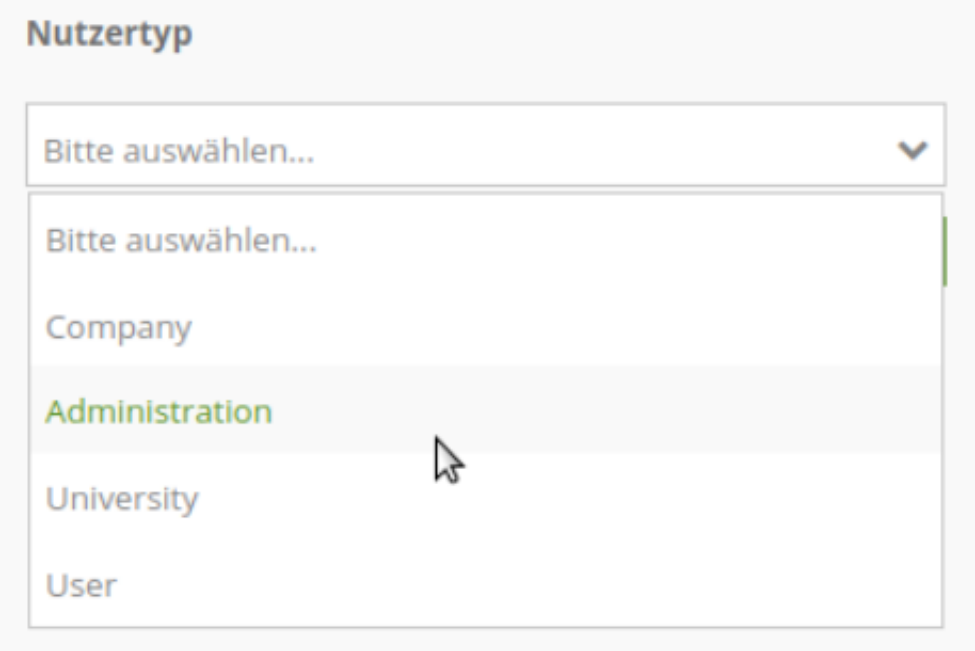
Die Auswahlmöglichkeiten werden im Bereich choices definiert. Dabei werden ein Wert (value) und ein Schlüssel (key) angegeben.

- **key**: Wird bei der Suchanfrage verwendet.
- **value**: Wird in der Auswahlbox angezeigt.

Note: Hinweis: Ab Mapbender 3.2 sollte die Angabe in der Reihenfolge `value: key` erfolgen, die Typdefinition lautet `type: Symfony\Component\Form\Extension\Core\Type\ChoiceType`.

---

type choice-Beispiel für ein Feld mit Auswahlmöglichkeiten:




```
type:
    type: Symfony\Component\Form\Extension\Core\Type\ChoiceType
↪            # box with selection options as dropdown list
    options:
        label: Nutzertyp
        required: false
```

(continues on next page)

(continued from previous page)

```
        placeholder: 'Bitte auswählen...'
        choices:
            Company: A
            Administration: B
            University: C
            User: D
            Sonstiges: E
    compare: exact
```

Vergleichsmodus

Für jedes Feld kann ein Vergleichsmodus bestimmt werden. Dieser wird beim Senden der Suchanfrage verwendet.

Folgende Vergleichsmodi werde unterstützt:

- **exact:** genauer Vergleich, Schlüssel = Wert (key = val),
- **iexact:** Vergleich, bei der Groß- / Kleinschreibung nicht unterschieden wird (case-insensitive),
- **like:** Standard, zweiseitiges 'like',
- **like-left:** linksseitiges 'like',
- **like-right:** rechtsseitiges 'like',
- **ilike**: zweiseitiges 'like', bei dem Groß- / Kleinschreibung nicht unterschieden wird (case-insensitive - *searchstring*),
- **ilike-left:** linksseitiges 'like', bei dem Groß- / Kleinschreibung nicht unterschieden wird (case-insensitive - *searchstring),
- **ilike-right:** rechtsseitiges 'like', bei dem Groß- / Kleinschreibung nicht unterschieden wird (case-insensitive - searchstring*).

Ergebnisausgabe

Im Bereich **results** erfolgt die Definition zum Aufbau der Trefferliste und das Styling der Treffer in der Karte.

```
results:
    view: table
    count: true
    exportcsv: true
    sortBy: name
    sortOrder: asc #desc
    zoomToResultExtent: true
    headers:
        gid: ID
        name: Name
        city: City/Town
    callback:
        event: click
        options:
            buffer: 10
```

(continues on next page)

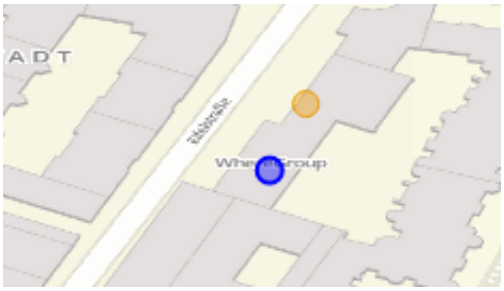
(continued from previous page)

```
            minScale: null
            maxScale: null
```

- **view**: Angabe `table`. Derzeit keine weiteren Optionen.
- **count**: Zeigt die Anzahl der Ergebnisse an (Standard: true).
- **exportcsv**: Erzeugt einen Button, der den Download der Trefferliste als CSV-Datei ermöglicht (Standard: false).
- **sortBy**: optional. Definition, ob eine Sortierung nach einem Attribut erfolgen soll. Angabe via attributname (Standard: keine Sortierung).
- **sortOrder**: optional. Definition, ob die Sortierung aufsteigend (asc) oder absteigend (desc) erfolgen soll (Standard: asc).
- **zoomToResultExtent**: Definition, ob direkt auf den Bereich der Ergebnisse gezoomt werden soll (Standard: false).
- **headers**: Definition der Ausgabespalten und Spaltenbeschriftung.
- **callback**: Definition der Aktion beim Klick auf einzelne Treffer.
- **event**: Angabe `click`. Derzeit keine weiteren Optionen.
- **buffer**: Zoom zum Treffer mit einem definierten Puffer.
- **minScale and maxScale**: Zoom zum Treffer in einem Maßstab zwischen minScale und maxScale.

Styling der Ergebnisse

Standardmäßig werden die Ergebnisse in der Karte in dem default-OpenLayers Style angezeigt.



Sie können diese Farbgebung überschreiben, indem Sie eine styleMap-Konfiguration definieren.

Die Konfiguration beinhaltet drei Stilkonfigurationen:

- **default**: Standardanzeige der Treffer,
- **select**: Anzeige bei Auswahl,
- **temporary**: Anzeige beim Mouseover.

```
results:
    [...]
    styleMap:
        default:
            strokeColor: '#00ff00'  # Umrandungsfarbe
            strokeOpacity: 1        # 1 - opak (keine Transparenz) 0 - voll transparent
            strokeWidth: 3          # Umrandungsbreite
```

(continues on next page)

(continued from previous page)

```
            fillColor: '#f0f0f0'    # Füllfarbe
            fillOpacity: 0          # Opazität der Füllung, 0 - voll transparent
            pointRadius: 6          # Größe des Punktsymbols
            label: ${name} - ${gid} # Attribut/e für die Beschriftung oder fester Text
            fontColor: "#ff0000"    # Beschriftungsfarbe
            fontSize: 30px          # Schriftgröße
            fontFamily: 'Arial, Courier New, monospace' # Schriftart
            fontWeight: bold
            labelOutlineColor: '#ffff00' # Schriftumrandungsfarbe
            labelOutlineWidth: 1    # Breite der Umrandung
            fontOffsetX: -18       # Versatz X-Achse
            fontOffsetY: -18       # Versatz Y-Achse
        select:
            strokeColor: '#0000ff'
            strokeOpacity: 1
            strokeWidth: 4
            fillColor: '#ff00ff'
            fillOpacity: 0.8
            pointRadius: 10
            label: ${name}
        temporary:
           strokeColor: '#0000ff'
           fillColor: '#0000ff'
           fillOpacity: 1
```

Wird Opacity auf 0 gesetzt (`fillOpacity: 0`), so wird die Fläche nicht gefüllt, sondern transparent dargestellt. In der default-Definition wird lediglich eine grüne Umrandung dargestellt (`strokeColor: '#00ff00'`). Die selektierten Objekte in diesem Beispiel werden mit der Farbe Lila gefüllt und sind mit leichter Opazität versehen. Sie verfügen über eine blaue Umrandung. Die temporäre Darstellung beim Mouse-Hover über ein Ergebnis erfolgt in blau.

Die styleMap-Angaben überschreiben Standardangaben. Daher müssen Sie lediglich die Angaben setzen, die Sie überschreiben möchten. Wenn Sie die Angaben weglassen wird der default-Stil verwendet.

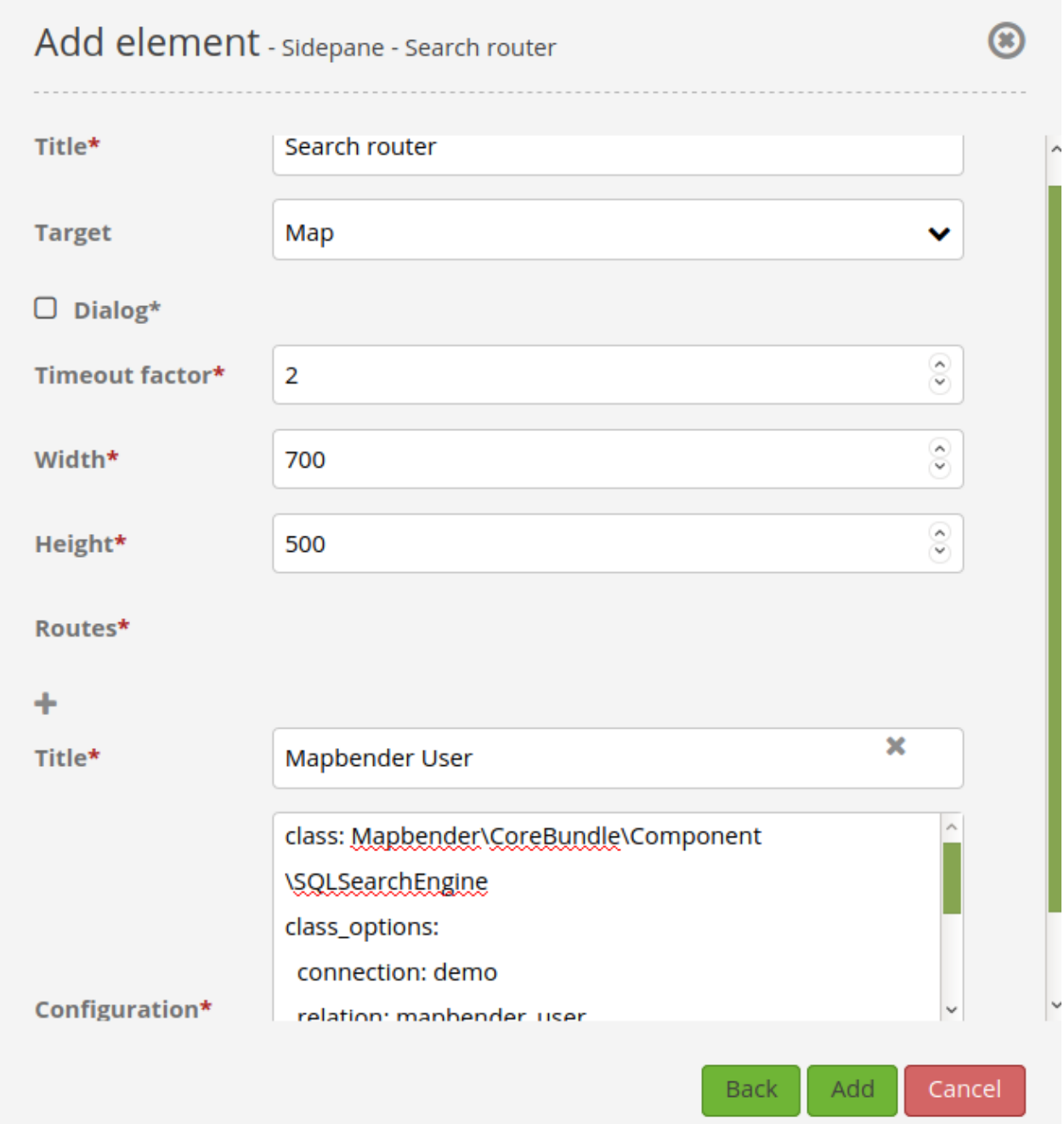
Note: Beachten Sie, dass die hexadezimalen Farbwerte in Anführungszeichen angegeben werden müssen, da das #-Zeichen ansonsten als Kommentar interpretiert wird.

Konfigurationsbeispiele

1. Beispiel

In diesem Beispiel wurde eine Suche für den “Mapbender User”-Layer konfiguriert. Die Suche wurde in die Sidepane über + in der Anwendung unter dem Layouts-Tab hinzugefügt.

Der Konfigurationsdialog zu diesem Konfigurationsbeispiel sieht wie folgt aus:



Der Elementitel (*Titel*) lautet Suchen, er wird in der Sidepane angezeigt. Da das Element in der Sidepane eingebunden wird und nicht als Button, wurde bei Dialog kein Haken gesetzt. Der *Timeout factor* wurde auf 2 gestellt. In den Feldern *Breite* und *Höhe* stehen zwar Werte, diese werden in der Anwendung jedoch nicht übergeben, da das Element in der Sidepane konfiguriert wird und diese die Werte nicht beachtet.

Es wurde über + bei *Routen* eine Suche eingebunden, die die Bezeichnung (*Titel*) Mapbender User trägt. In *Konfiguration* wurde die yaml-Konfiguration des Elements eingefügt. Vollständig lautet diese:

```
class: Mapbender\CoreBundle\Component\SQLSearchEngine
class_options:
  connection: geodata_db                                    # die Datenbank, auf die das
↪Element zugreift
```

(continues on next page)

(continued from previous page)

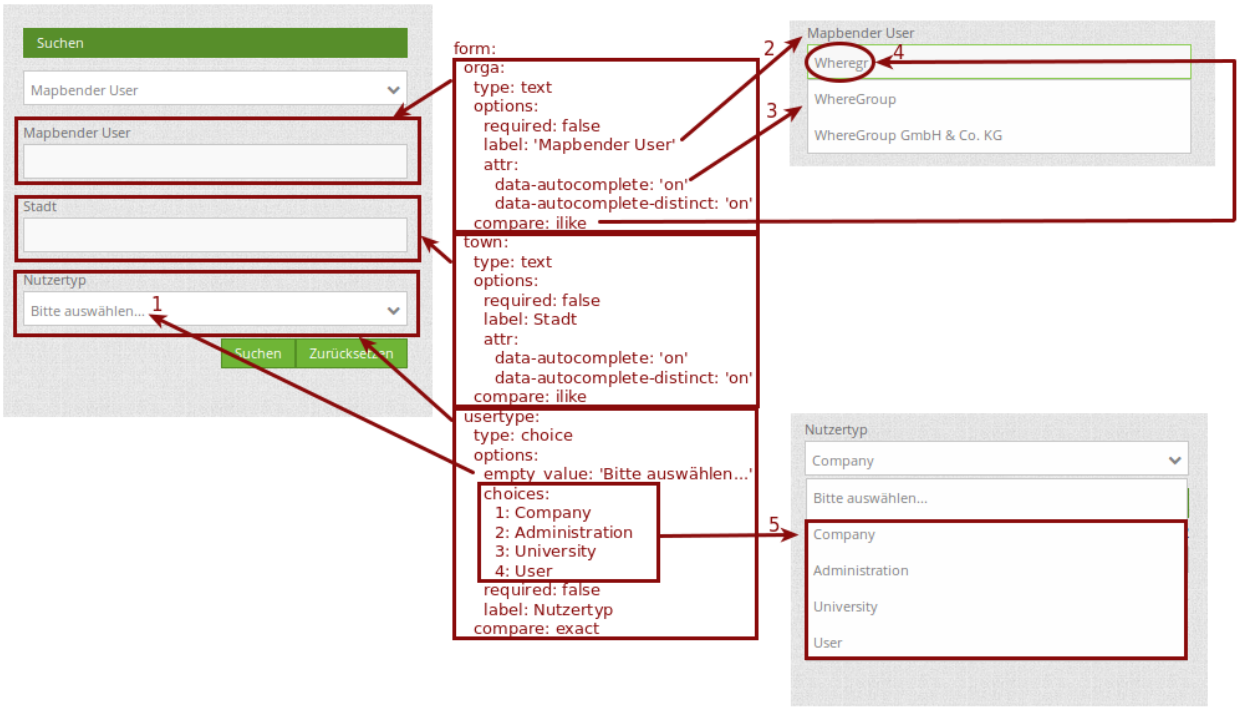
```
  relation: mapbender_user                                    # die Tabelle, auf die das
↪Element zugreift
  attributes:                                                 # entspricht den Tabellenspalten
↪in der Datenbank, die angesprochen werden sollen
    - gid
    - orga
    - town
    - usertype
  geometry_attribute: the_geom                                # Definition der Geometrie Spalte
form:                                                         # ab hier beginnt die
↪Konfiguration des Formulars
  orga:                                                       # Feld für die Suche nach dem
↪Namen des Mapbender Users
    type: Symfony\Component\Form\Extension\Core\Type\TextType
    options:
      required: false                                         # kein Pflichtfeld
      label: 'Mapbender User'                                 # Überschrift über dem Feld
      attr:                                                   # zusätzlich definierbare
↪Attribute
        data-autocomplete: 'on'                               # automatische Vervollständigung
↪des eingetippten Suchbegriffs
        data-autocomplete-distinct: 'on'
    compare: ilike                                            # Vergleichsmodus
  town:                                                       # Feld für die Suche nach der
↪Stadt
    type: Symfony\Component\Form\Extension\Core\Type\TextType
    options:
      required: false                                         # kein Pflichtfeld
      label: Stadt                                            # Überschrift über dem Feld
      attr:
        data-autocomplete: 'on'
        data-autocomplete-distinct: 'on'
    compare: ilike
  usertype:                                                   # Feld für die Suche nach dem
↪Nutzertyp
    type: Symfony\Component\Form\Extension\Core\Type\ChoiceType
    options:
      placeholder: 'Bitte auswählen...'                       # Text, der angezeigt wird, bevor
↪etwas ausgewählt wurde
      choices:                                                # die Auswahlmöglichkeiten;
↪werden wie folgt angegeben: "Eintrag in der Spalte der Datenbank": "Angezeiger Name in der Dropdown-
↪Liste"
        1: Company
        2: Administration
        3: University
        4: User
      required: false                                         # kein Pflichtfeld
      label: Nutzertyp                                        # Überschrift über dem Feld
    compare: exact                                            # Vergleichsmodus
results:                                                      # Konfiguration der
↪Ergebnisanzeige
  view: table                                                 # Tabelle ausgeben
  count: true                                                 # Anzahl der Ergebnisse anzeigen
  headers:                                                    # Titel der Spalte; werden wie
↪folgt angegeben: Spaltenname in der Datenbank: Bezeichnung der Spalte in der Suchausgabe der Anwendung
    gid: ID
```

(continues on next page)

(continued from previous page)

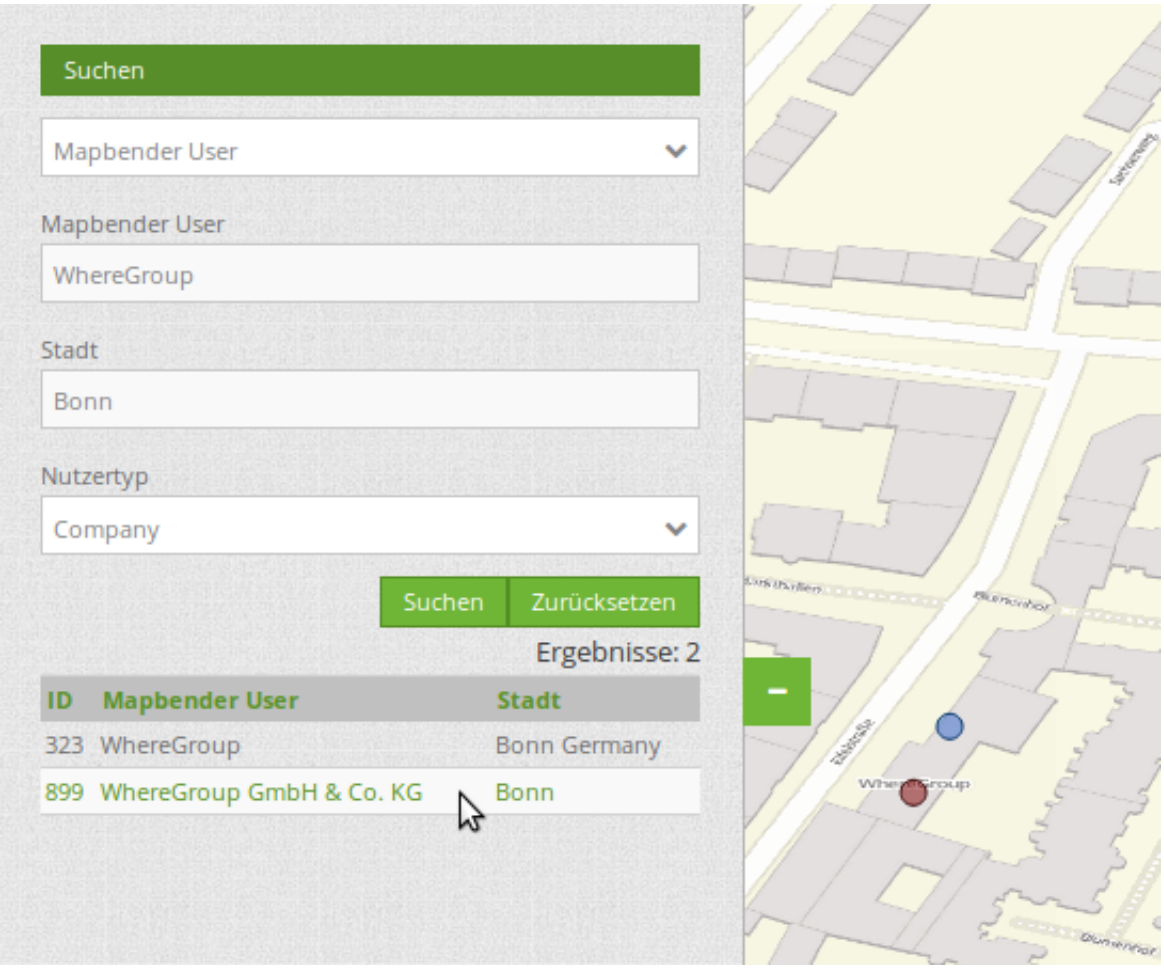
```
    orga: 'Mapbender User'
    town: Stadt
  callback:
    event: click                                          # bei Klicken wird das Element
↪selektiert
    options:
      buffer: 10
      minScale: null
      maxScale: 10000
  styleMap:                                               # Styling der Punkte in der Karte
    default:                                              # Styling aller angezeigten Punkte
      strokeColor: '#003366'
      strokeOpacity: 1
      fillColor: '#3366cc'
      fillOpacity: 0.5
    select:                                               # Styling des selektierten Objekts
      strokeColor: '#330000'
      strokeOpacity: 1
      fillColor: '#800000'
      fillOpacity: 0.5
    temporary:
      strokeColor: '#0000ff'
      fillColor: '#0000ff'
      fillOpacity: 1
```

Auf dieser Abbildung wird gezeigt, welche Auswirkungen die vorgenommenen Konfigurationen in der yaml-Definition auf das Suchformular haben:

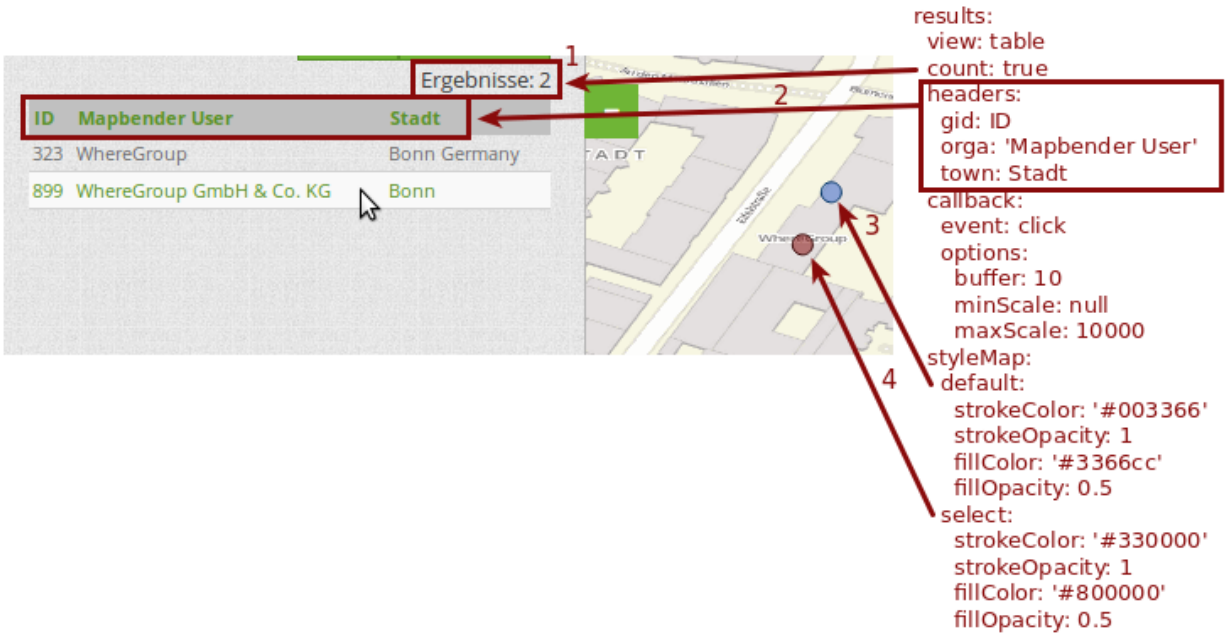


Dargestellt ist der Ausschnitt der yaml-Definiton, der das Formular konfiguriert. Tabellenspalten *orga*, *town* und *usertype* werden im Formular verwendet und sind jeweils als die Felder *Mapbender User*, *Stadt* und *Nutzertyp* eingebunden. *Mapbender User* und *Stadt* sind jeweils vom Typ Text. *Nutzertyp* gibt hingegen Auswahlmöglichkeiten in Form einer Dropdown-Liste vor.

Eine vollständig ausgefüllte Suche nach dem Mapbender User WhereGroup, in der Stadt Bonn, des Nutzertyps Company und deren Ergebnis sieht wie folgt aus:



Auf dieser Abbildung wird gezeigt, welche Auswirkungen die vorgenommenen Konfigurationen in der yaml-Definition auf die Anzeige der Ergebnisse haben:



Auf dieser Abbildung ist lediglich die Konfiguration der Ergebnisse angezeigt. Die Anzahl der Ergebnisse wird aufgrund von *count: true* angezeigt. Anschließend werden die Spaltentitel unter *headers* definiert. Hier wird zuerst die Bezeichnung der Spalte in der Tabelle angegeben, sodass definiert wird, auf welche Tabellenspalte sich die Ergebnisanzeige bezieht. Nach dem Doppelpunkt wird angegeben, welcher Titel in der Anwendung angezeigt werden soll.

Im Block *styleMap* wird das Styling der Punkte vorgenommen. Der Block *default* bezieht sieht dabei auf alle Punkte und der Block *select* nur auf das ausgewählte Objekt.

Da keines dieser Felder ein Pflichtfeld ist, kann die Suchabfrage auch nur mithilfe eines Feldes erfolgen.

2. Beispiel

Das folgende Beispiel baut auf dem Deutschen Geographischen Namenskatalog im Maßstab 1:250.000 des Bundesamtes für Kartographie und Geodäsie auf. Die Daten wurden in die Tabelle `gn250_p` der Datenbank `gisdb` kopiert und eignen sich gut für eine Ortssuche. Die Daten haben viele verschiedene Spalten, u.a.:

- **id**: Die ID des Datensatzes,
- **name**: Der Name des Datensatzes,
- **kreis**: Der Landkreis (nicht für jeden Datensatz vorhanden),
- **oba_wert**: Die Art des Features (z.B. Bahnhof, Museum, etc.).

Beispiel einer Suchkonfiguration im `Konfiguration` Bereich:

```
class: Mapbender\CoreBundle\Component\SQLSearchEngine
class_options:
  connection: geodata_db
  relation: gn250_p
  attributes:
    - id
    - name
```

(continues on next page)

(continued from previous page)

```
    - kreis
    - oba_wert
  geometry_attribute: geom
form:
  name:
    type: Symfony\Component\Form\Extension\Core\Type\TextType
    options:
      required: true
    compare: ilike
results:
  view: table
  count: true
  exportcsv: true
  sortBy: name
  sortOrder: asc
  zoomToResultExtent: true
  headers:
    id: ID
    name: Name
    kreis: Landkreis
    oba_wert: Art
  callback:
    event: click
    options:
      buffer: 10
      minScale: 500
      maxScale: 2500
```

YAML-Definition

Diese Vorlage kann genutzt werden, um das Element in einer YAML-Anwendung einzubinden.

```
target: map                                          # ID des Kartenelements
asDialog: true                                       # true, Erebniswiedergabe in einem Dialogfeld
timeoutFactor:  3                                    # Timeout-Faktor (multipliziert mit autcomplete
↪Verzögerung) um die Autovervollständigung zu verhindern, nachdem eine Suche gestartet wurde
height: 500                                          # Höhe des Dialogs
width: 700                                           # Breite des Dialogs
routes:                                              # Sammlung von Suchrouten
    demo_polygon:                                    # für Maschinen lesbarer Name
   class: Mapbender\CoreBundle\Component\SQLSearchEngine  #  Suchmaschine, die verwendet werden soll
   class_options:                                    # Diese werden an die Suchmaschine weitergegeben
       connection: geodata_db                        # search_db, DBAL Verbindungsname, der benutzt
↪werden soll, benutzen Sie ~ für default
       relation: polygons                            # Verbindungsauswahl, Unterabfragen können
↪verwendet werden
       attributes:
           - gid                                     # Liste von Spalten auswählen, expressions are
↪possible
           - name
           - type
           - city
       geometry_attribute: geom                      # Name der Geometriesplate, die genutzt werden
↪soll. Achtung: Projektion muss mit Projektion des map-Elements übereinstimmen
   form:                                             # Einstellungen für das Suchformular
```

(continues on next page)

(continued from previous page)

```
        name:                                   # Feldname, Spaltenname der genutzt werden soll
            type: Symfony\Component\Form\Extension\Core\Type\TextType    # Eingabefeld, normalerweise
↪Text oder Zahlen
            options:                            # Einstellungen für das Eingabefeld
                required: false                 # HTML5 benötigte Attribute
                label: Name                     # benutzerdefinierte Beschriftung eingeben, sonst
↪wird die Beschriftung von dem Feldnamen abgeleitet
                attr:                           # HTML5 benötigte Attribute
                    data-autocomplete: on       # Attribut, um Autovervollständigung zu aktivieren
                    data-autocomplete-distinct: on    # Attribut, dass Autovervollständigung aktiviert
↪aber unterscheiden lässt
                    data-autocomplete-using: type     # komma separierte Liste von anderen
↪Eingabefeldern, in denen WHERE Angaben für die Autovervollständigung gemacht werden
            compare: ilike                      # Siehe unten für weitere Vergleichsformen
        city:
            type: Symfony\Component\Form\Extension\Core\Type\TextType
            options:
                required: false
                label: Ort/Stadt
            compare: ilike
        type:
            type: Symfony\Component\Form\Extension\Core\Type\ChoiceType
            options:
                placeholder: 'Bitte auswählen...'
                required: false
                choices:
                    A Company: A
                    B Administration: B
                    C University: C
                    D User: D
                    E Somthing else: E
    results:
        view: table                             # Ansicht der Ergebnisse, Ausgabe z.B. als Tabelle
        count: true                             # Anzahl der Treffer anzeigen
        exportcsv: true
        headers:                                # Bezeichnung der Tabellenüberschriften und der
↪entsprechenden Ergebnisspalten
            gid: ID                             # Spaltenname -> Überschrift
            name: Name
            type: Type
            city: Ort/Stadt
        callback:                               # Was beim Klick und Mauszeiger halten passiert
            event: click                        # Ergebnisliste (click oder mouseover)
            options:
                buffer: 10                      # Puffert die Geometrieergebnise (Karteneinheiten)
↪vor dem Zoomen
                minScale: ~                     # Maßstabsbegrenzung beim Zoomen, ~ für keine
↪Begrenzung
                maxScale: ~
        styleMap:
            default:
                strokeColor: '#00ff00'
                strokeOpacity: 1
                fillOpacity: 0
            select:
                strokeColor: '#ff0000'
```

(continues on next page)

(continued from previous page)

```
            fillColor: '#ff0000'
            fillOpacity: 0.4
        temporary:
            strokeColor: '#0000ff'
            fillColor: '#0000ff'
            fillOpacity: 1
```

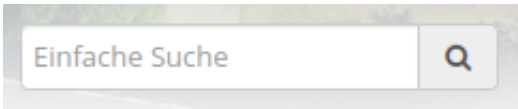
3.5.2 Einfache Suche (SimpleSearch)

Die Einfache Suche bietet eine Einfeldsuche oder Schlagwortsuche. Die Suchanfrage wird dabei an einen Suchdienst übermittelt. Als Such-Server können Apache Solr, Nominatim, photon oder ein OGC API Features Dienst verwendet werden.

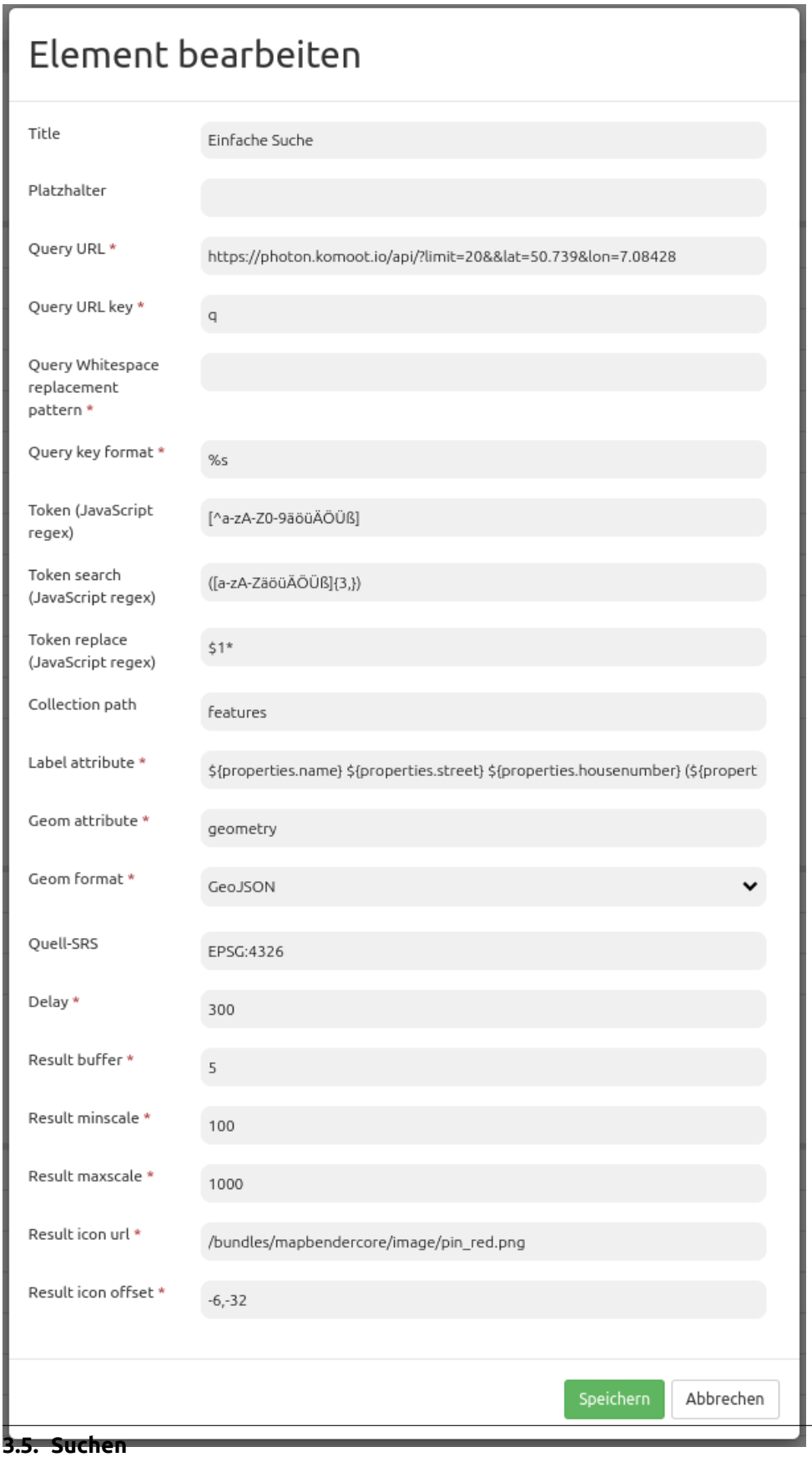
Es wird ein Suchfeld angeboten, das in jede Region eingebunden werden kann. SimpleSearch sendet den eingegebenen Suchbegriff an eine konfigurierbare URL und empfängt JSON-formatierte Daten, welche eine Beschriftung und Geometrieattribute für jeden Eintrag beinhalten.

Die Geometriedaten können in WKT oder in GeoJSON-Format codiert werden.

Es kann frei konfiguriert werden, welche Informationen in der Trefferauswahl angezeigt werden sollen.



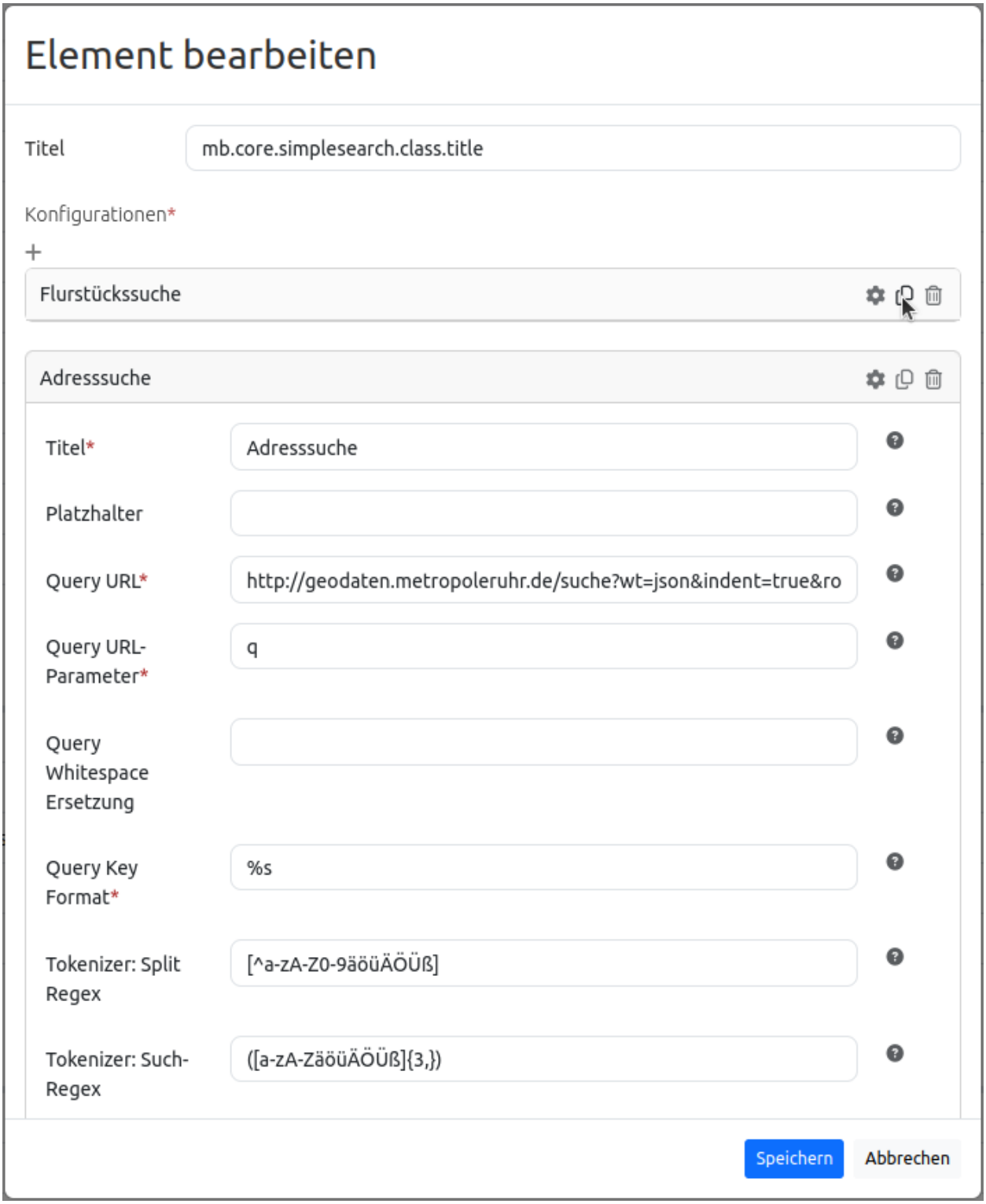
Konfiguration



- **Title:** Titel des Elements. Dieser wird in der Layouts Liste angezeigt und ermöglicht, mehrere Button-Elemente voneinander zu unterscheiden. Der Titel wird außerdem neben dem Button angezeigt, wenn "Beschriftung anzeigen" aktiviert ist.
- **Position:** Positionierung (nur bei Nutzung im Kartenbereich)
- **Platzhalter:** Text, der im Suchfeld angezeigt wird, wenn dieses leer ist. Wird playeholder nicht gesetzt, wird der Titel verwendet.
- **Query URL:** Solr bzw. Nominatim URL, an die der eingegebene Suchbegriff gesendet wird (z.B. `http://localhost:8080/solr/core0/select?wt=json&indent=true`).
- **Query URL key:** Der Suchparameterschlüssel, der angehängt wird (z.B. `q`).
- **Query Whitespace replacement pattern:** Muster zum Austausch von Leerzeichen.
- **Query key format:** Einfaches Suchformat (z.B. `%s`).
- **Token search/replace (JavaScript regex): Tokenizer spaltet/sucht/ersetzt regexp.**
  - **Token**, z.B.: `[^a-zA-Z0-9äöüÄÖÜß]`
  - **Token search**, z.B..: `([a-zA-ZäöüÄÖÜß]{3,})`
  - **Token replace**, z.B..: `$1*`
- **Collection path:** Dies kann ein Attributspfad sein, der vom Abfrageergebnis extrahiert wird (z.B. `response.docs`).
- **Label attribut:** Attribut oder mehrere Attribute , die als Ergebnis angezeigt werden sollen.
- **Geom attribut:** Attributname der Geodaten (z.B. `geom`).
- **Geom format:** Geodatenformat, kann WKT oder GeoJSON sein (z.B. `WKT`).
- **Quell-SRS:** EPSG-Code des primären Koordinatenbezugsystems (z.B. `EPSG:25832`)
- **Delay:** Automatische Vervollständigungs-Verzögerung (z.B. `300`).
- **Result buffer:** Puffert das Objekt (in Karteneinheiten) vor dem Zoomen (z.B. `10`).
- **Result minscale/maxscale:** Maßstabsbegrenzung beim Zoomen (z.B. `1000` und `5000`). ~ wenn keine Begrenzung gewünscht wird.
- **Result icon url:** Symbol, das zur Trefferanzeige verwendet werden soll (z.B. `/bundles/mapbendercore/image/pin_red.png`).
- **Result icon offset:** Versatz x und y des Symbols (z.B. `-6,-38` für das Stecknadel-Icon).

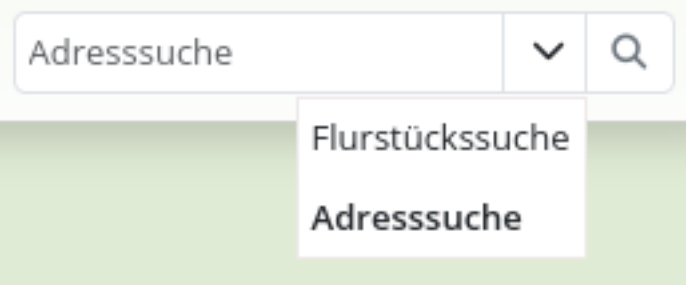
Duplizieren einer Suchkonfiguration

Das Element unterstützt mehrere Suchkonfigurationen. Es ist möglich, ein Suchformular im Backend-Konfigurationsfenster mithilfe des Kopieren-Buttons zu duplizieren:



Die verschiedenen Suchkonfigurationen sind außerdem via Drag and drop sortierbar.

In der Anwendung können die eingebundenen Konfigurationen über eine Dropdown-Auswahl angewählt werden:



Flexible Konfigurationsmöglichkeiten für label_attribute

Das JSON enthält unterschiedliche Informationen. Es können auch mehrere der Attribute als Ergebnis ausgegeben werden.

```
label_attribute: label
```

Die Definition bezieht sich relativ zur Angabe unter *Collection path*. Zwischen den einzelnen Attributen kann zusätzlicher Text angebenen werden.

```
label_attribute:  '${properties.address.city} ${properties.address.road} ${properties.address.house_
↪number}'
```

```
label_attribute:  'Ort: ${properties.address.city}: ${properties.address.road} - ${properties.address.
↪house_number}'
```

YAML-Definition

```
placeholder: 'Bitte Suchbegriff eingeben....'                                    # Text, der im
↪Suchfeld angezeigt wird, wenn dieses leer ist. Wird playeholder nicht gesetzt, wird der Titel
↪verwendet.
anchor: 'right-bottom'                                                           # Positionierung
↪(nur bei Nutzung im Kartenbereich). Optionen: 'left-top', 'right-top', 'left-bottom', 'right-bottom'
query_url: http://example.com/solr/core/0/select?wt=json&indent=true&rows=8      # Solr URL (z.B.
↪``http://localhost:8080/solr/core0/select?wt=json&indent=true``) oder Nominatim URL.
query_key: q                                                                     # Der
↪Suchparameterschlüssel, der angehängt wird
query_ws_replace:                                                                # Pattern zum
↪Austausch von Leerzeichen.
query_format: '%s'                                                               # Einfaches
↪Suchformat.
token_regex: [^a-zA-Z0-9äöüÄÖÜß]                                                 # Tokenizer split
↪regexp.
token_regex_in: ([a-zA-ZäöüÄÖÜß]{3,})                                            # Tokenizer search
↪regexp.
token_regex_out: '$1*'                                                           # Tokenizer replace
↪regexp.
collection_path: response.docs                                                   # Es kann ein
↪Attributspfad sein, der vom Abfrageergebnis extrahiert wird.
label_attribute: label                                                           # Attributname, der
↪für die Trefferausgabe genutzt wird
geom_attribute: geom                                                             # Name des Attributs
↪der Geometriedaten
```

(continues on next page)

(continued from previous page)

```
geom_format: WKT                                                    # Geodatenformat,
↪kann WKT oder GeoJSON sein
sourceSrs: 'EPSG:25832'                                             # Projektion in der
↪Die Daten ausgeliefert werden
delay: 300
result:                                                             # Automatische
↪Vervollständigungs-Verzögerung. 0
  buffer: 50                                                        # Puffert den
↪Treffer in Karteneinheiten vor dem Zoomen
  minscale: 1000                                                    # Maßstabsbegrenzung
↪beim Zoomen, ~ für keine Begrenzung
  maxscale: 5000                                                    # Maßstabsbegrenzung
↪beim Zoomen, ~ für keine Begrenzung
  icon_url: /bundles/mapbendercore/image/pin_red.png                # Marker, der zur
↪Trefferanzeige verwendet werden soll
  icon_offset: -6,-38                                               # Versatz x und y
↪des Symbols
```

Einrichtung von Solr:

- **Download**: https://solr.apache.org/downloads.html
- **Dokumentation**: https://solr.apache.org/guide/solr/latest/index.html
- **Tutorial**: https://solr.apache.org/guide/solr/latest/getting-started/solr-tutorial.html

Einrichtung von Nominatim:

- **Download**: https://nominatim.org/release-docs/latest/admin/Installation/
- **Dokumentation**: https://nominatim.org/release-docs/latest/

Einrichtung von photon:

photon ist ein Open Source Geocoder für OpenStreetMap-Daten.

- **Download & Dokumentation**: https://github.com/komoot/photon

Konfiguration OGC API Features

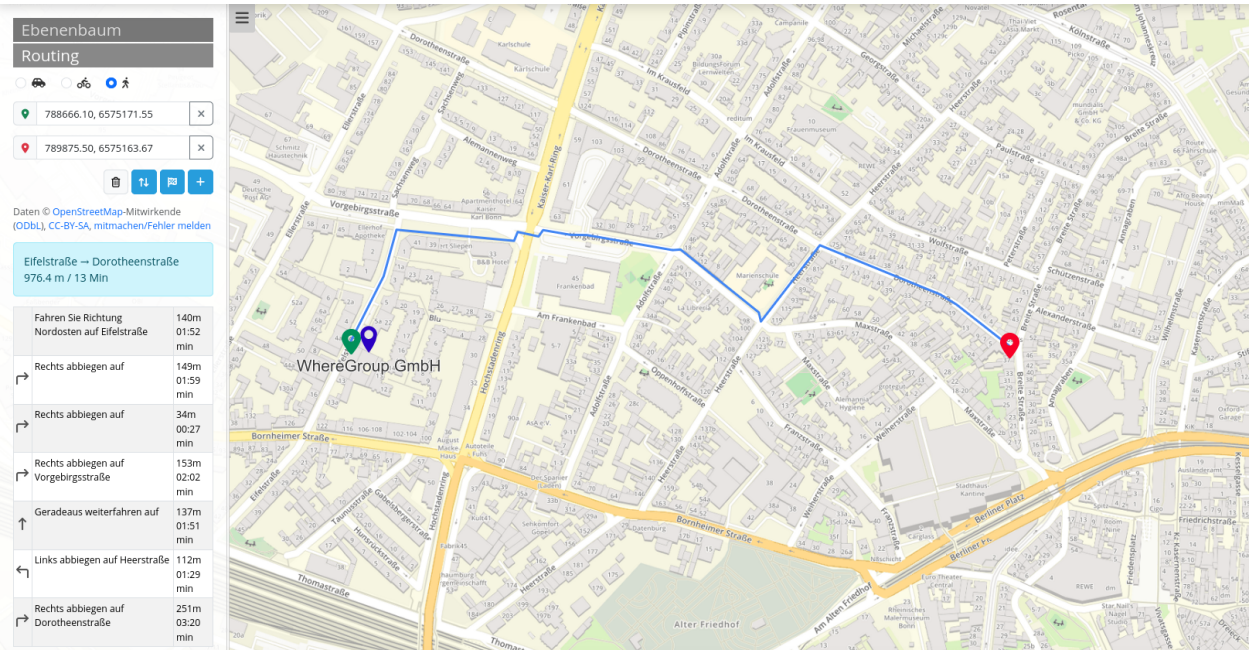
Diverse Projekte unterstützen OGC API Features wie QGIS, GeoServer, pygeoapi.

HTTP Callbacks

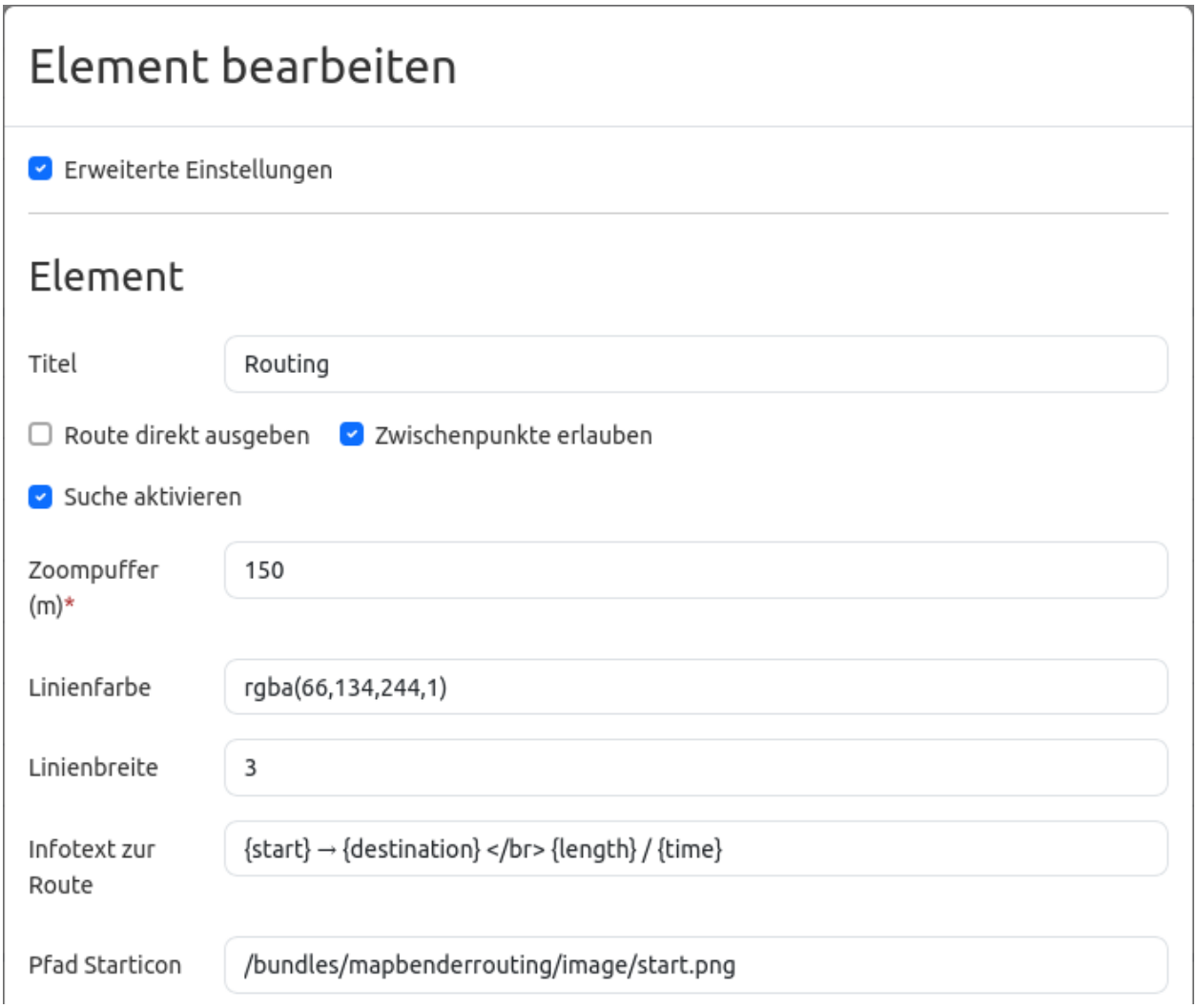
- /search: Proxy-Element, welches die konfigurierbare URL abfragt. Im Entwicklungsmodus wird die endgültige Abfrage-URL zum einfachen Debugging als ein x-mapbender-simplesearch-url Header zurückgegeben.

3.5.3 Routing

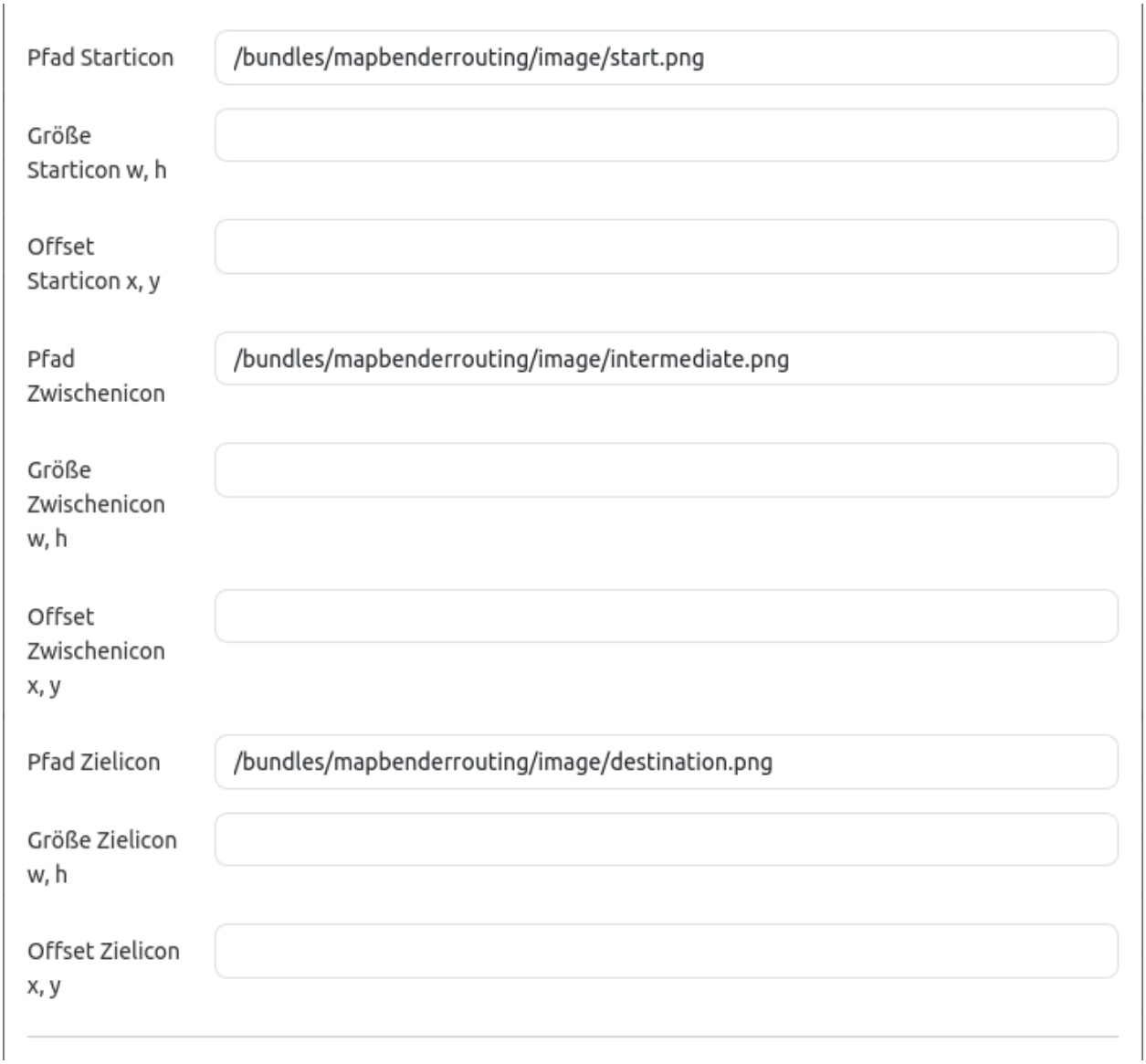
Das Routing-Element fügt einer Anwendung ein Streckenführung-Werkzeug hinzu. Nach Angabe von Start, Ziel und ggf. Zwischenpunkten wird eine geeignete Streckenführung in der Karte angezeigt. Zusätzlich können Informationen zur Strecke ausgegeben werden.



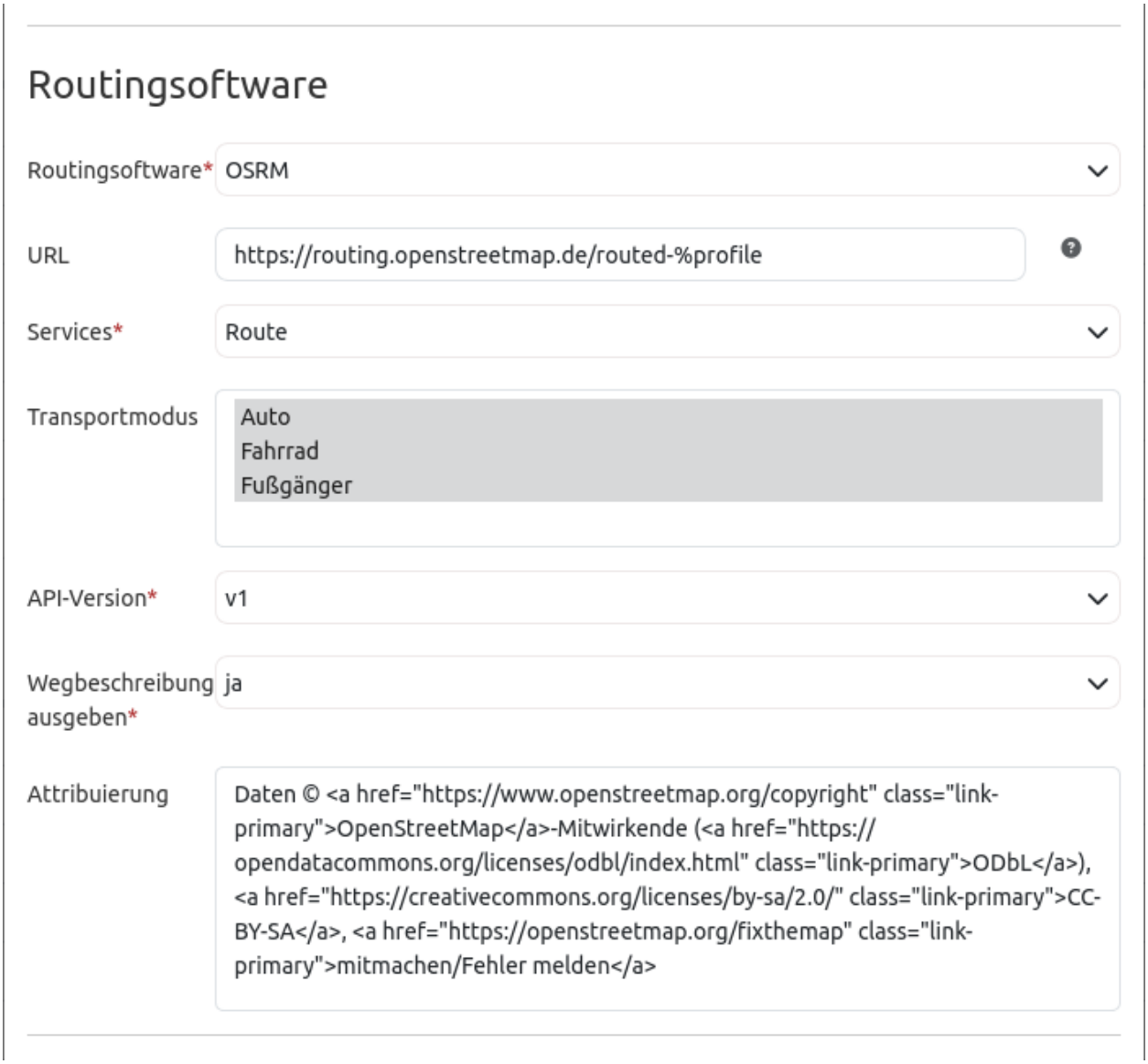
Konfiguration



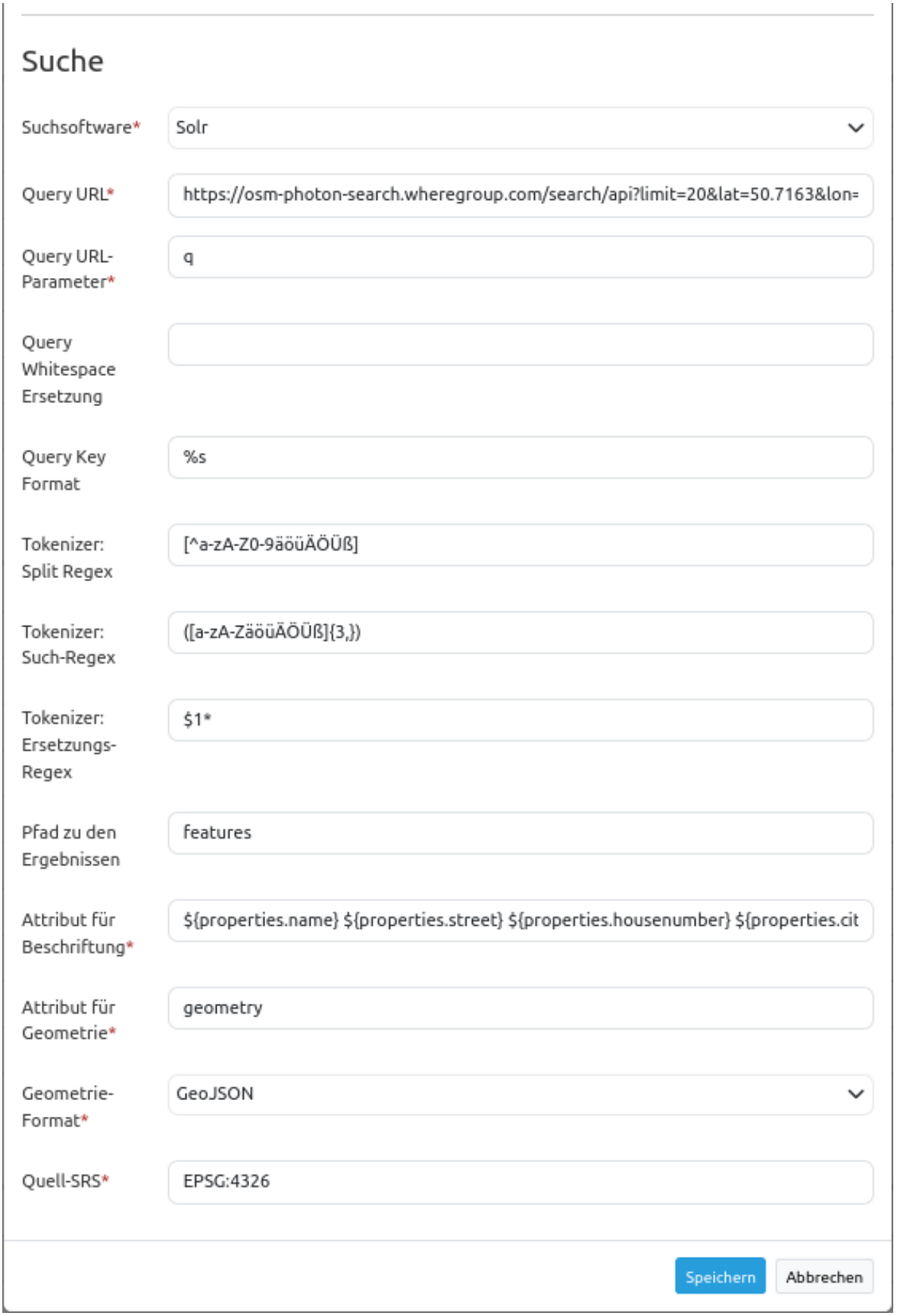
- **Erweiterte Einstellung:** Ermöglicht es weitere Einstellungen vorzunehmen (Standard: false).
- **Titel:** Titel des Elements.
- **Route direkt ausgeben:** Konfiguration zum Deaktivieren/Aktivieren des automatischen Routings ohne Interaktion des Nutzers (Standard: false).
- **Zwischenpunkte erlauben:** Konfiguration zum Deaktivieren/Aktivieren der Zwischenpunkten (Standard: false).
- **Suche aktivieren:** Konfiguration zum Deaktivieren/Aktivieren der Suchoption (Standard: false).
- **Zoompuffer(m):** Definition eines Zoompuffers für die Ergebnisanzeige in Meter (Standard: 0).
- **Linienfarbe:** Anpassen der Lineinfarbe und Deckkraft per rgba Standard (Standard: rgba(66, 134, 244, 1)).
- **Linienbreite:** Anpassen der Linienbreite (Standard: 3).
- **Infotext zur Route:** Option einen Infotext zur Route zu implementieren (Standard: {start} □ {destination} </br> {length} will take {time}).



- **Pfad Starticon:** Anpassen des Starticons (Standard: /bundles/mapbenderrouting/image/start.png).
- **Pfad Zwischenicon:** Anpassen des Zwischenicons (Standard: /bundles/mapbenderrouting/image/intermediate.png).
- **Pfad Zielicon:** Anpassen des Zielicons (Standard: /bundles/mapbenderrouting/image/destination.png).
- **Größe Icons:** Die Größe der verschiedenen Icons ist anpassbar.
- **Offset icons:** Das Offset der verschiedenen Icons ist anpassbar.



- **Routingsoftware:** Auswählen der Routingsoftware (OSRM, GraphHopper, PgRouting, Trias).
- **URL:** Setzen der URL-Adresse für die Routingsoftware oder wählen einer Variable, z.B. https://routing.openstreetmap.de/routed-%profile (Standard: https://).
- **Services:** Auswahl aus verschiedenen Services (Standard: Route).
- **Transportmodus:** Auswählen der Routing-Profile (Auto, Fahhrad, Füßgänger). Auch die Auswahl mehrerer ist möglich (Standard: false).
- **API-Version:** Version der API bestimmen (Standard: v1).
- **Wegbeschreibung ausgeben:** Auswählen ob es eine Wegbeschreibung geben soll oder nicht (Standard: nein).
- **Attributierung:** Angabe eines Quellenverweises (Standard: Siehe oben im Konfigurations Screenshot zur Routingsoftware).


Suche
Suchsoftware*
Solr
Query URL*
https://osm-photon-search.wheregroup.com/search/api?limit=20&lat=50.7163&lon=
Query URL-Parameter*
q
Query Whitespace Ersetzung
Query Key Format
%s
Tokenizer: Split Regex
[^a-zA-Z0-9äöüÄÖÜß]
Tokenizer: Such-Regex
([a-zA-ZäöüÄÖÜß]{3,})
Tokenizer: Ersetzungs-Regex
$1*
Pfad zu den Ergebnissen
features
Attribut für Beschriftung*
${properties.name} ${properties.street} ${properties.housenumber} ${properties.cit
Attribut für Geometrie*
geometry
Geometrie-Format*
GeoJSON
Quell-SRS*
EPSG:4326
Speichern
Abbrechen

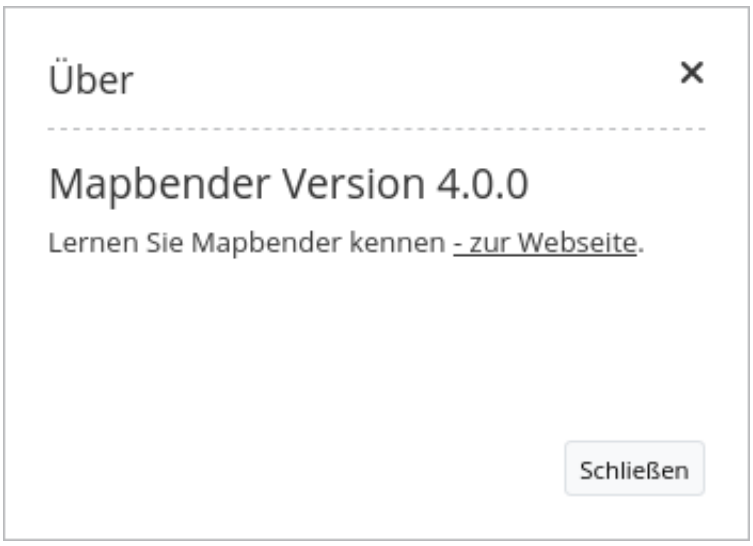
- **Suchsoftware:** Auswählen des Such-Dienstes (derzeit nur Solr).
- **Query URL:** Setzen der URL-Adresse für die Suchsoftware.
- **Query URL-Parameter:** Der Suchparameterschlüssel, der angehängt wird (Standard: q).
- **Query Whitespace Ersetzung:** Setzen von Paramteren zum Ersetzung im Such-Therm.
- **Query Key Format:** Setzen des Such-Formats (Standard: %s).
- **Token Split-/Such-/Ersetzungs-Regex (JavaScript regex): Tokenizer spaltet/sucht/ersetzt regexp.**
  - **Token Split**, z.B.: `[^a-zA-Z0-9äöüÄÖÜß]`
  - **Token Such**, z.B..: `([a-zA-ZäöüÄÖÜß]{3,})`
  - **Token Ersetzung**, z.B..: `$1*`
- **Pfad zu den Ergebnissen:** Setzen des Pfads, der vom Abfrageergebnis extrahiert wird (Standard: response.docs).
- **Attribut für Beschriftung:** Attribut oder mehrere Attribute , die als Ergebnis angezeigt werden sollen (Standard: label).
- **Attribut für Geometrie:** Attributname der Geometrie (Standard: geom).
- **Geometrieformat:** Geometrieformat, kann WKT oder GeoJSON sein (Standard WKT).
- **Quell-SRS:** EPSG-Code des primären Koordinatenbezugsystems (Standard EPSG:4326).

YAML-Definition

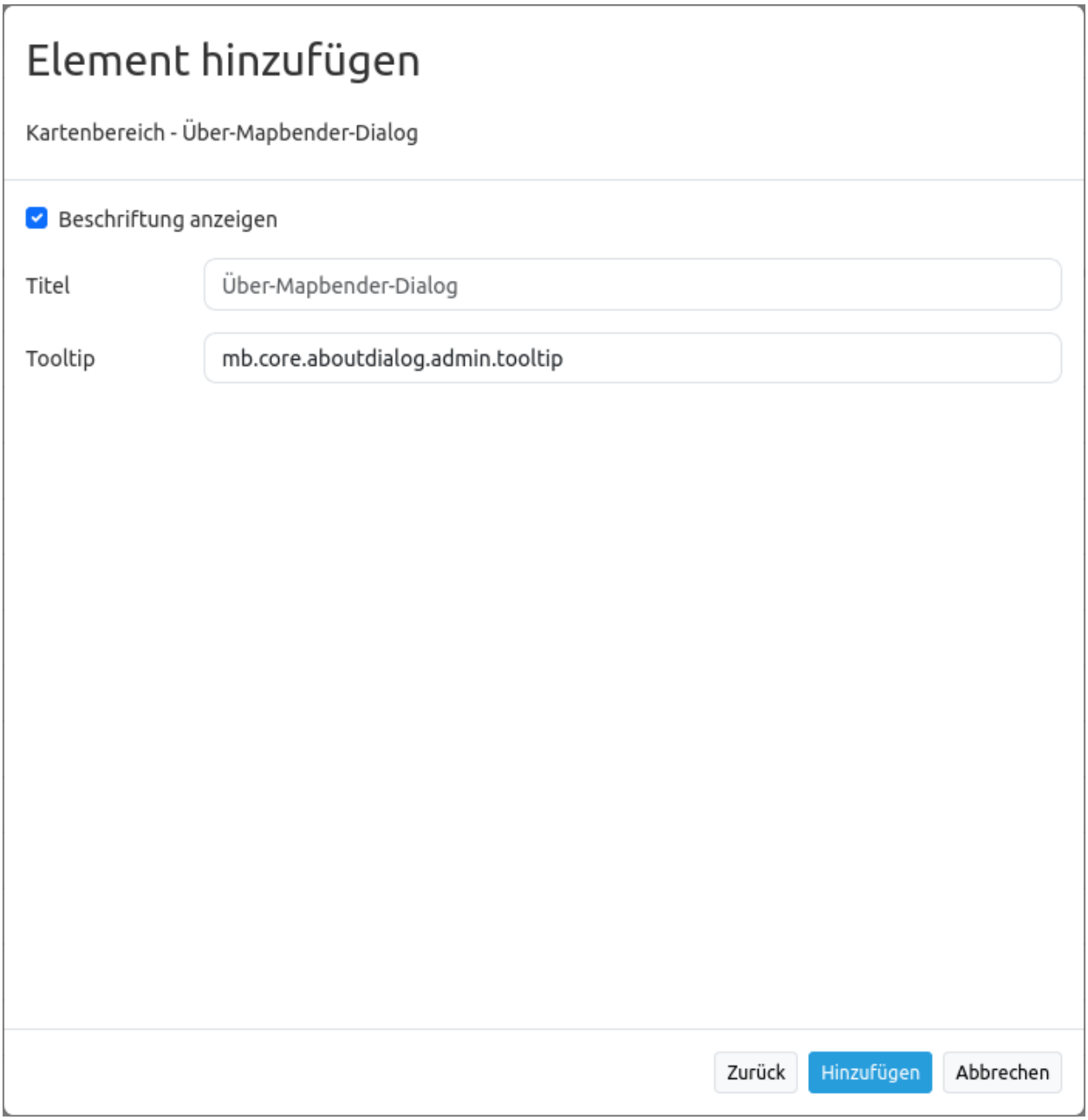
3.6 Verschiedenes

3.6.1 Über-Mapbender-Dialog (About Dialog)

Dieses Element erstellt einen *Button*, der einen Dialog mit der aktuellen Mapbender Version anzeigt. Der Button kann in die obere Werkzeugleiste und in die Fußzeile eingefügt werden.



Konfiguration



- **Beschriftung anzeigen:** Schaltet Text neben dem About Dialog Button ein/aus (Standard: true).
- **Titel:** Text, der neben dem About Dialog Button angezeigt wird.
- **Tooltip:** Text, der erscheint, wenn der Mauszeiger längere Zeit über dem Button gehalten wird.

YAML-Definition

Diese Vorlage kann genutzt werden, um das Element in einer YAML-Anwendung einzubinden.

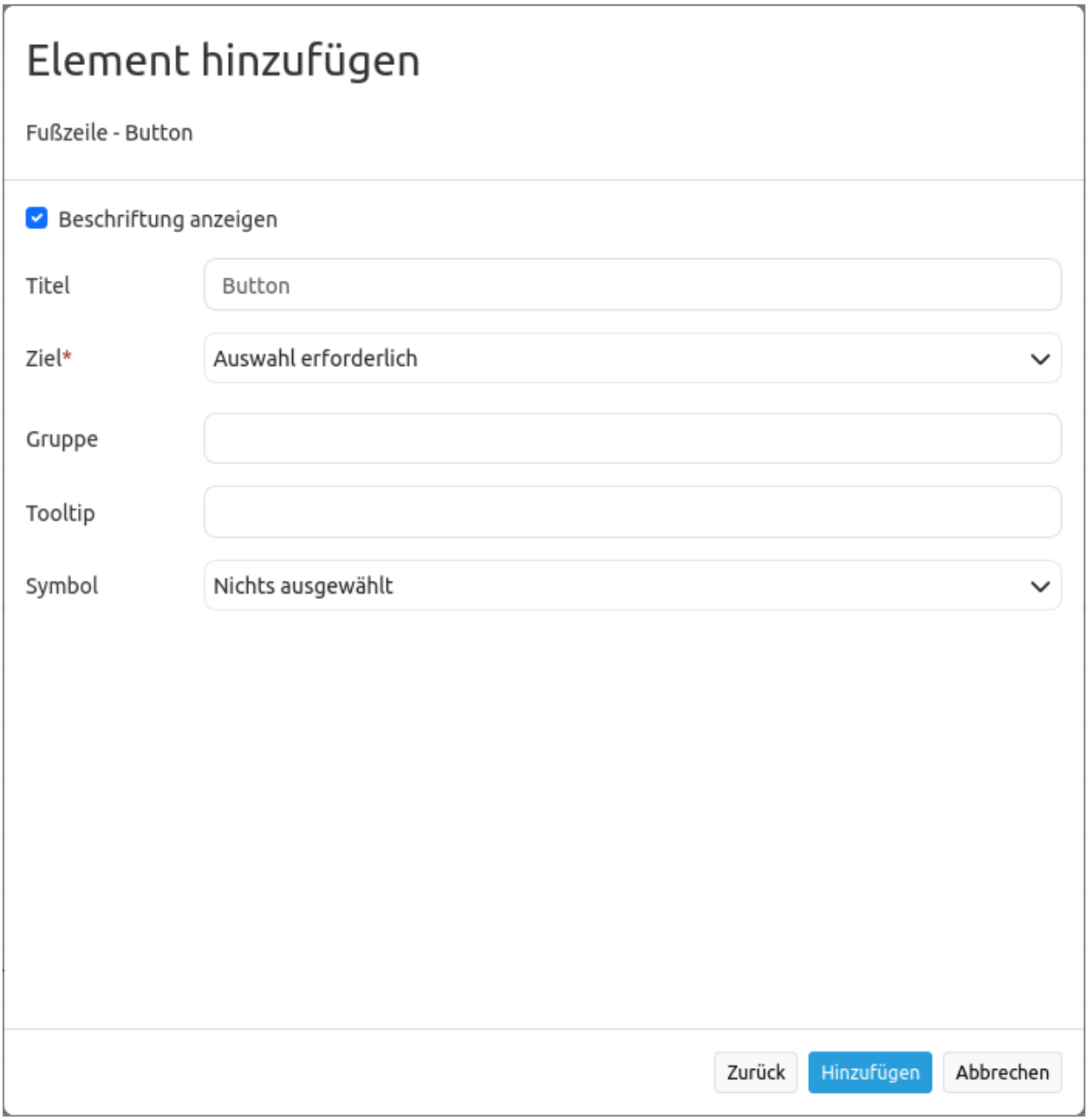
```
title: 'Über Mapbender'     # Text, der neben dem About Dialog Button angezeigt wird
tooltip: 'Über Mapbender'   # Text des Tooltips
label: true                  # false/true, um den Button zu beschriften (Standard: true)
icon: 'abouticon'            # Symbol für den Button
```

3.6.2 Button

Dieses Element stellt ein Button-Modul bereit. Einige Elemente wie die *Legende (Legend)*, *Ebenenbaum (Layertree)*, *Information (FeatureInfo)*, *Linien-/Flächenmessung (Line/Area Ruler)* und der *Druck (PrintClient)* benötigen einen Button, um einen Dialog anzuzeigen oder um aktiviert zu werden, wenn das Element in den Kartenbereich oder die Fußleiste eingebunden wurde.

Buttons können optional gruppiert werden, sodass nur ein Button in der Gruppe aktiviert ist. Dies wird im Gruppen-Parameter eingestellt. Es kann außerdem ein Button definiert werden, der sich auf eine Webseite oder ein Script bezieht und bei Aktivierung zu diesem weiterleitet. Bei dem Parameter *Ziel* stehen nur Funktionen zur Auswahl, die vorher in der Anwendung unter dem Reiter *Layouts* entweder in den Kartenbereich oder die Fußleiste eingebunden wurden.

Konfiguration



- **Beschriftung anzeigen:** Schaltet die Beschriftung des Buttons an/aus (Standard: an).
- **Titel:** Titel des Elements. Dieser wird in der Layouts Liste angezeigt und ermöglicht, mehrere Button-Elemente voneinander zu unterscheiden. Der Titel wird außerdem neben dem Button angezeigt, wenn "Beschriftung anzeigen" aktiviert ist.
- **Ziel:** Zielelement (Titel), das bei Anklicken des Buttons ausgelöst wird.
- **Gruppe:** Hiermit kann das Element einer Gruppe hinzugefügt werden. Aus dieser Gruppe können nicht mehrere Buttons gleichzeitig aktiviert sein. Wird ein anderer Button aus der Gruppe ausgewählt, wird der vorher ausgewählte automatisch deaktiviert.
- **Tooltip:** Text, der angezeigt wird, wenn der Mauszeiger eine längere Zeit über dem Element ver-

weilt.

- **Symbol:** Symbol des Buttons, basierend auf einer CSS-Klasse.

Icons

Für einige Symbole können zwei verschiedene Icon-Typen ausgewählt werden.

- Ein Symbol basierend auf einer Grafik (z.B. "About"),
- Ein Symbol basierend auf einer Schrift (z.B. "About (Font Awesome)").

Letztere basieren auf einem IconSet, das mit dem Mapbender als Modul ausgeliefert wird. Wir empfehlen die Verwendung der Symbole aus dieser Bibliothek.

Hint: Es ist auch möglich, ein Icon-Set zu deaktivieren und/oder andere Icons zu verwenden. Weitere Informationen finden Sie unter *Icons anpassen*.

Mehr Informationen zu Icons unter:

- https://github.com/mapbender/icons
- https://rawgit.com/mapbender/icons/master/demo.html

Konfigurationsbeispiele:

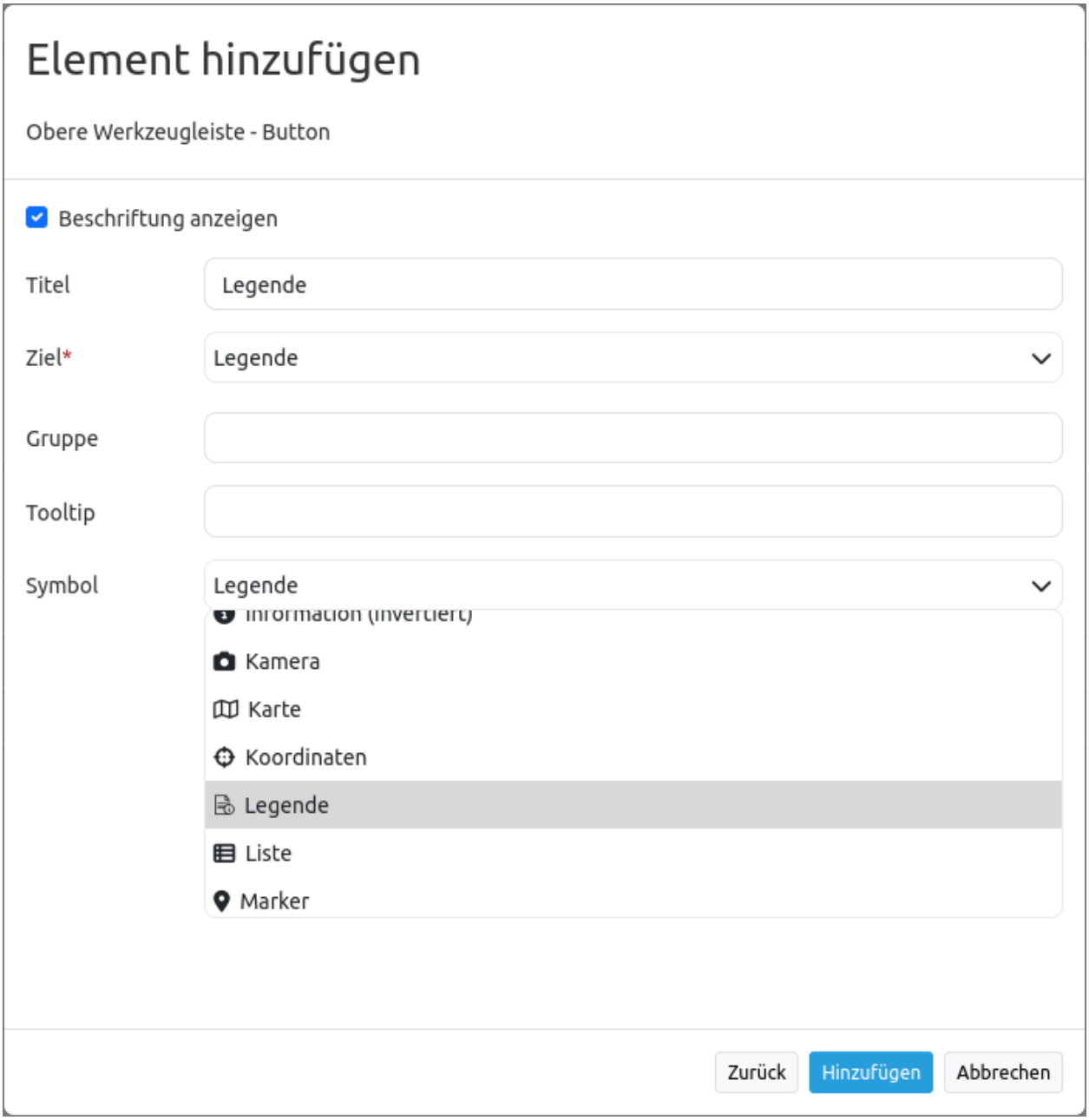
Je nach Ziel der Anwendung werden unterschiedliche Buttons benötigt, die verschiedene Funktionen bieten. Diese können nach Bedarf und Wunsch integriert werden. Buttons können für Features eingebunden werden, die vorher im Kartenbereich konfiguriert wurden. Beispielsweise können die Legende oder die Linien- und/oder Flächenmessung über Buttons angesprochen werden:

Button für die Legende

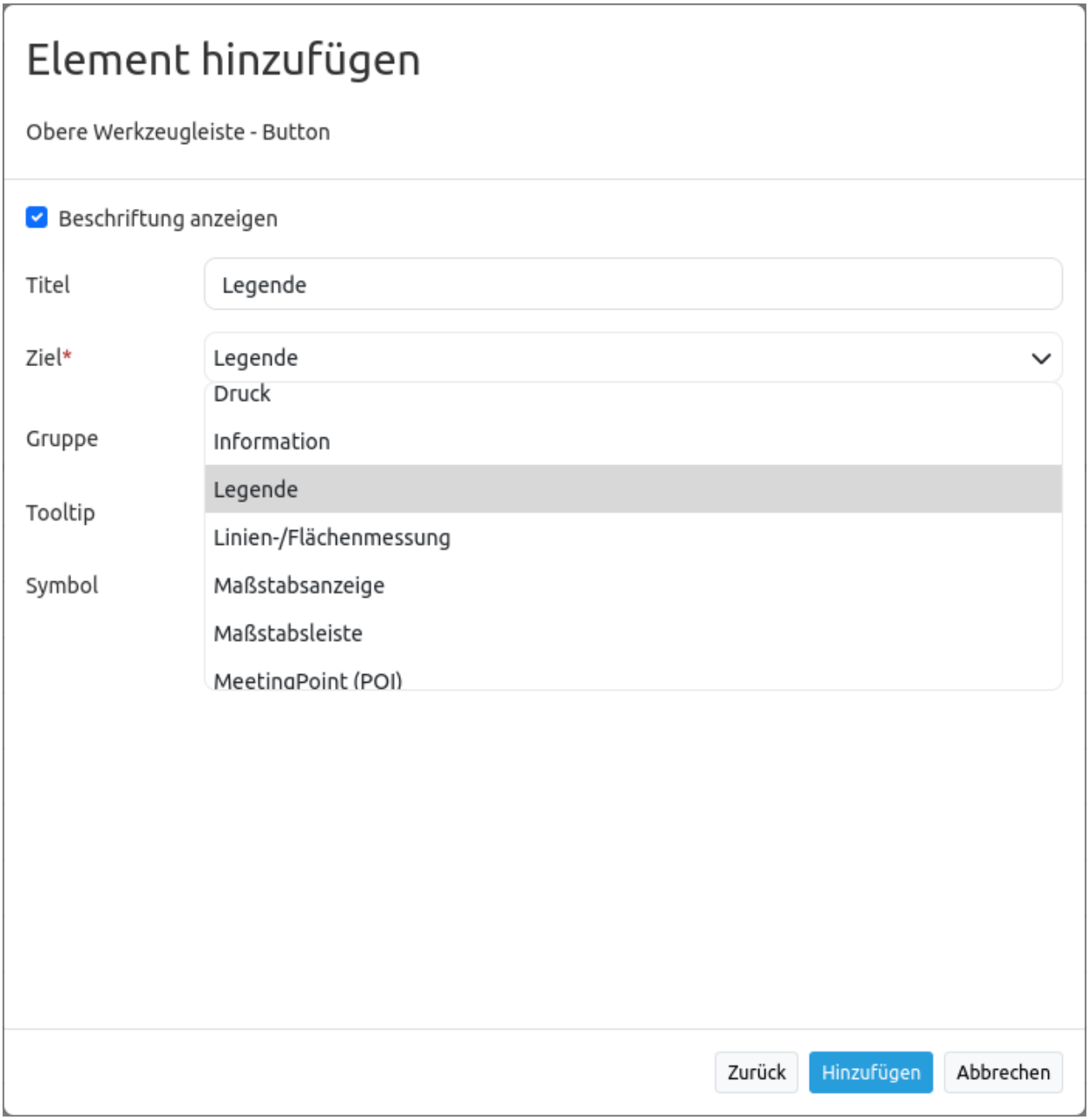
Für Karten sind Legenden sehr hilfreich, da sich so die Betrachter der Karte über den Inhalt informieren können. Die Legende ist in diesem Anwendungsbeispiel im Kartenbereich eingebunden. Wie eine Legende konfiguriert wird, wird in der Dokumentation unter *Legende (Legend)* beschrieben. Der Button für eine Legende wird wie folgt eingebunden:

Zuerst muss über + in der Anwendung (unter Layouts, Obere Werkzeugleiste) das Element Button ausgewählt werden.

Nach Auswahl des Elements Button öffnet sich der Dialog "Element hinzufügen – Button". Hier werden die Einstellungen zur Konfiguration des Buttons vorgenommen. Die Bezeichnung des Buttons zum Öffnen der Legende wird im Feld *Titel* eingetragen und lautet hier "Legende". Im Beispiel wurde kein Text für "Tooltip" definiert. Das heißt, es erscheint kein Text, wenn die Maus über den Button innehält. Als Icon können nun eine Vielzahl an Möglichkeiten ausgewählt werden. In diesem Fall wird die Option "Legend" gewählt.



Nun wird im Feld *Ziel* gewählt, welche vorher definierte Funktion mit dem Button angesprochen werden soll. Die Liste, die dort als Dropdown-Menü erscheint, beinhaltet alle Funktionen, die vorher im Kartenbereich, in der Sidepane oder Fußzeile konfiguriert wurden. Da hier der Button für die Legende konfiguriert wird, muss ebenfalls die Option “Legend” gewählt werden.



Das Feld “Gruppe” wird für diesen Fall leer gelassen. Der Button sieht in der Mapbender-Anwendung wie folgt aus:

Da im Konfigurationsdialog zum Button Element bei *Beschriftung anzeigen* ein Haken gesetzt war, wird die Beschriftung neben dem Icon angezeigt. Ist dieser Haken nicht aktiviert, sieht der Button wie folgt aus:

Wäre hier im Feld *Symbol* statt “Legend” “Liste” ausgewählt und die Beschriftung aktiviert worden, sähe

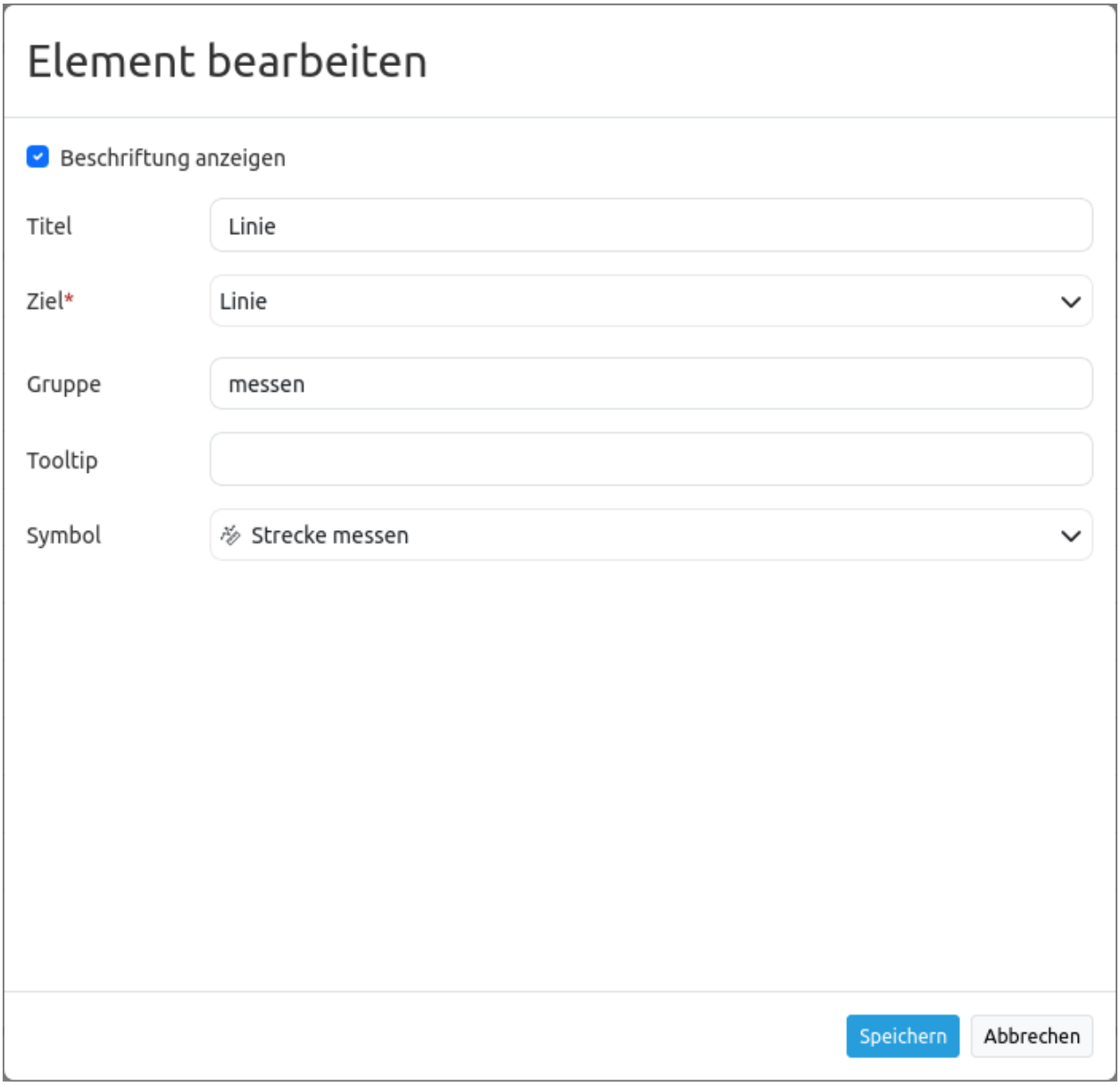
der Button wie folgt aus:

Nun öffnet sich beim Klick auf den Button die Legende in einem Dialogfenster.

Button für Linien- und Flächenmessung

Auch die Funktionen der Linien- und Flächenmessung können über Buttons in eine Anwendung eingebunden werden. Auch in diesem Fall ist es notwendig, dass diese Funktionen schon in Kartenbereich, Sidepane oder Fußzeile konfiguriert sind.

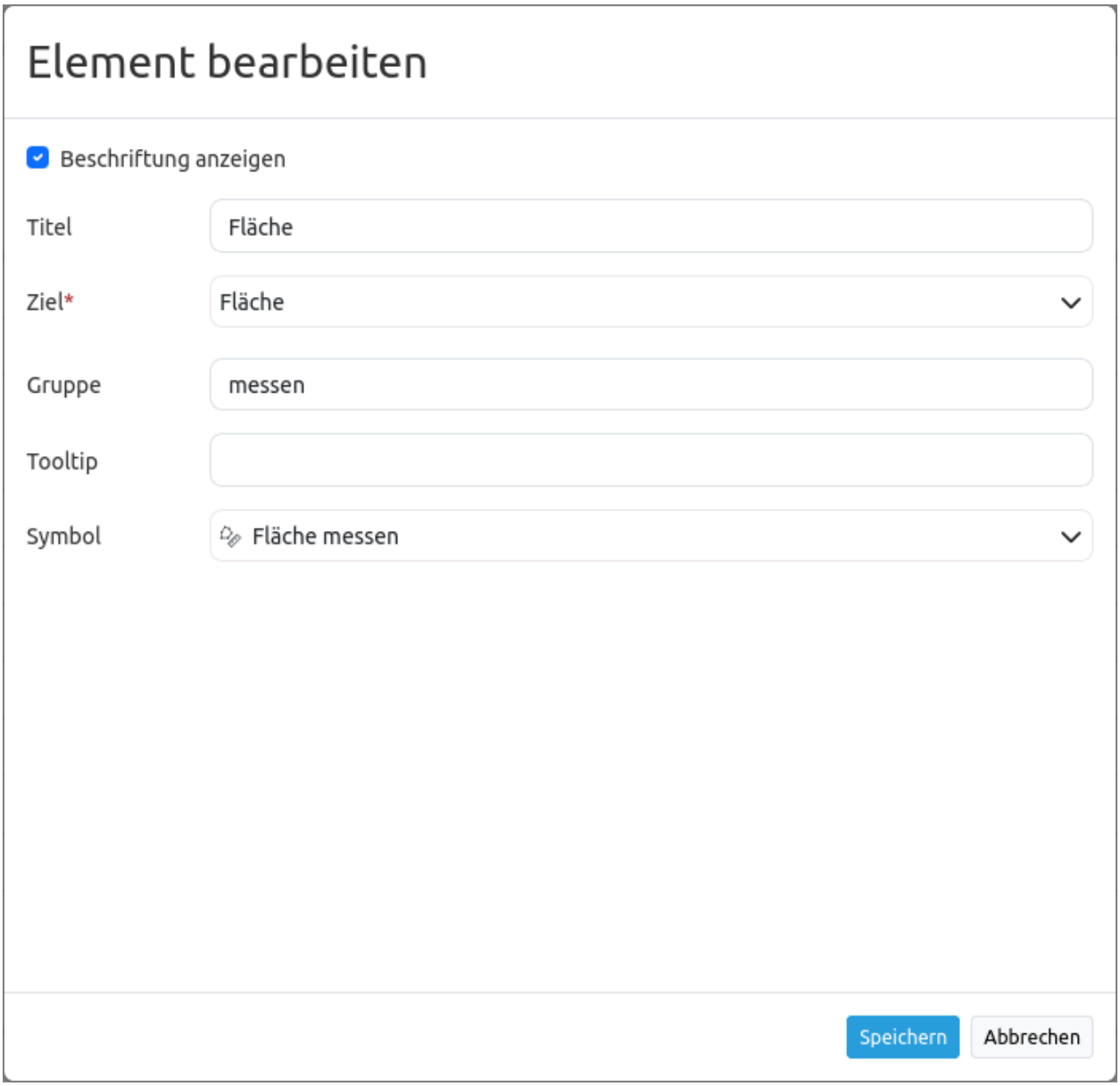
In diesem Konfigurationsbeispiel sollen beide Buttons einer Gruppe zugeordnet werden, so dass entweder Linien oder Flächen gemessen werden können, jedoch nicht beides gleichzeitig.



Im Anwendungsbeispiel ist der *Titel* des Buttons “Linie”. Als *Target* wird das vorher im Kartenbereich erstellte Element “Linie” eingebunden. Um die Gruppierung mit der Flächenmessung möglich zu machen, wird im Feld *Gruppe* ein Gruppenname vergeben. Hier lautet die Bezeichnung der Gruppe “messen”. Dieser Gruppenname wird analog auch bei dem Button für die Flächenmessung eingetragen. Als *Symbol* wird “Strecke messen” gewählt.

Das Element “Linie” wurde mithilfe der Funktion Linien-/Flächenmessung erstellt und als Linienmessung konfiguriert. Wie das Element Linien-/Flächenmessung konfiguriert wird, wird in der Dokumentation unter *Linien-/Flächenmessung (Line/Area Ruler)* beschrieben.

Der Button für die Flächenmessung wird analog eingebunden. Der Dialog der Konfiguration des Buttons sieht im Konfigurationsbeispiel wie folgt aus:



Zu beachten ist besonders das Feld *Gruppe* mit dem Namen der Gruppe. Dieser muss mit dem Eintrag im Feld *Gruppe* des Buttons “Linie” übereinstimmen muss. Beide Buttons können in der Anwendung wie folgt aussehen:

Ist der Button Linienmessung aktiv, sieht er wie folgt aus:

Sobald nun auf den Button Flächenmessung geklickt wird, wird die Funktion Linienmessung beendet und die Funktion Flächenmessung aktiviert.

YAML-Definition

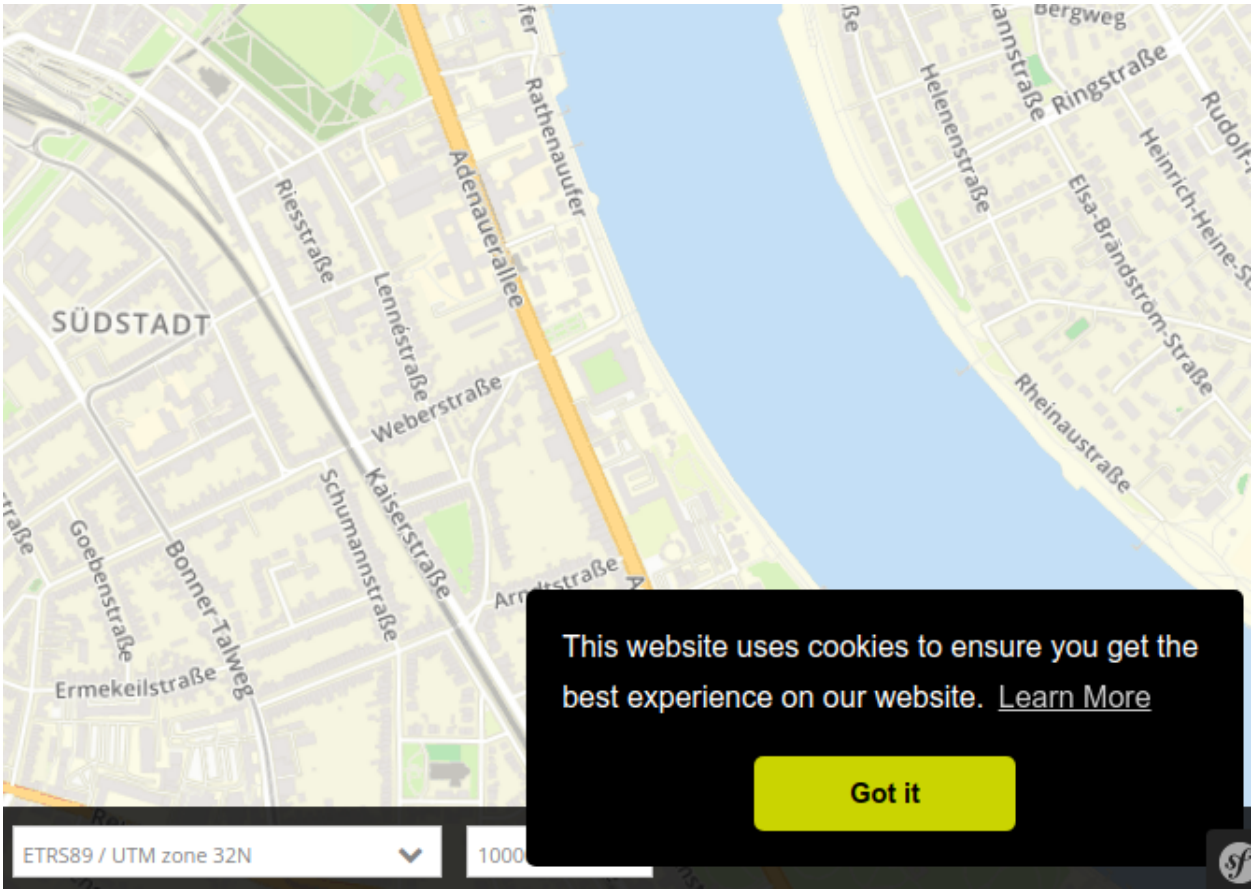
Diese Vorlage kann genutzt werden, um das Element in einer YAML-Anwendung einzubinden.

```
title:      # Titel
tooltip:    # Text des Tooltips
icon: ~     # Symbol verwendete CSS Klasse
label: true # false/true, um den Button zu beschriften. Der Standardwert ist true.
target: ~   # Titel (Id) des Zielelements
click:      # bezieht sich auf eine Webseite oder ein Skript, z.B.: http://mapbender.org
group: ~    # Gruppe, in die der Button eingefügt werden soll. Nur ein Button pro Gruppe kann
↪aktiviert sein.
action: ~   # Methode, die aufgerufen wird, wenn der Button aktiviert wird.
deactivate: ~ # Methode, die aufgerufen wird, wenn der Button deaktiviert wird
```

3.6.3 Cookie-Banner

Anwendungen unterstützen die Anzeige eines Cookie-Banners, welches in der Mapbender-Konfigurationsdatei `parameters.yaml` eingerichtet werden kann. Wir nutzen dazu den Code von Cookie Consent, ohne dass dabei ein Aufruf ins Internet gestartet wird.

Das Banner erscheint in einer beliebigen Anwendung beim ersten Aufruf.



Sobald die Mitteilung akzeptiert wurde, taucht das Banner so lange nicht auf, bis die Cookies im Webbrowser gelöscht worden sind. Mapbender ist abhängig von Cookies und dort wird die PHP-Session hinterlegt.

Konfiguration

Die Konfiguration findet in der *parameters.yaml*-Datei (unterhalb von *config/*) statt und gilt für die gesamte Mapbender-Instanz. Dazu muss der Parameter `mapbender.cookieconsent:` mit dem Wert `true` oder `false` übergeben werden. Ist der Abschnitt nicht vorhanden oder der Parameterwert `false` gesetzt, wird das Banner nicht angezeigt.

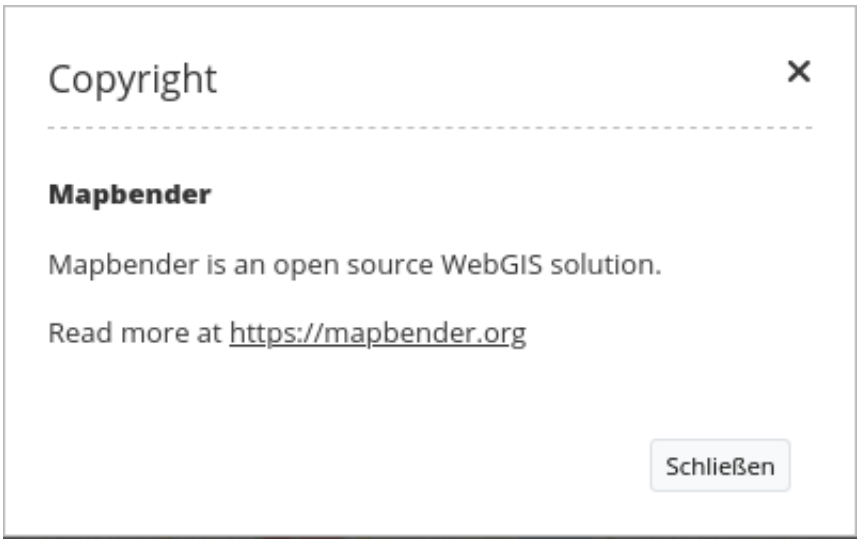
Beispiel:

```
#Mapbender Cookie Consent Message
mapbender.cookieconsent: true
```

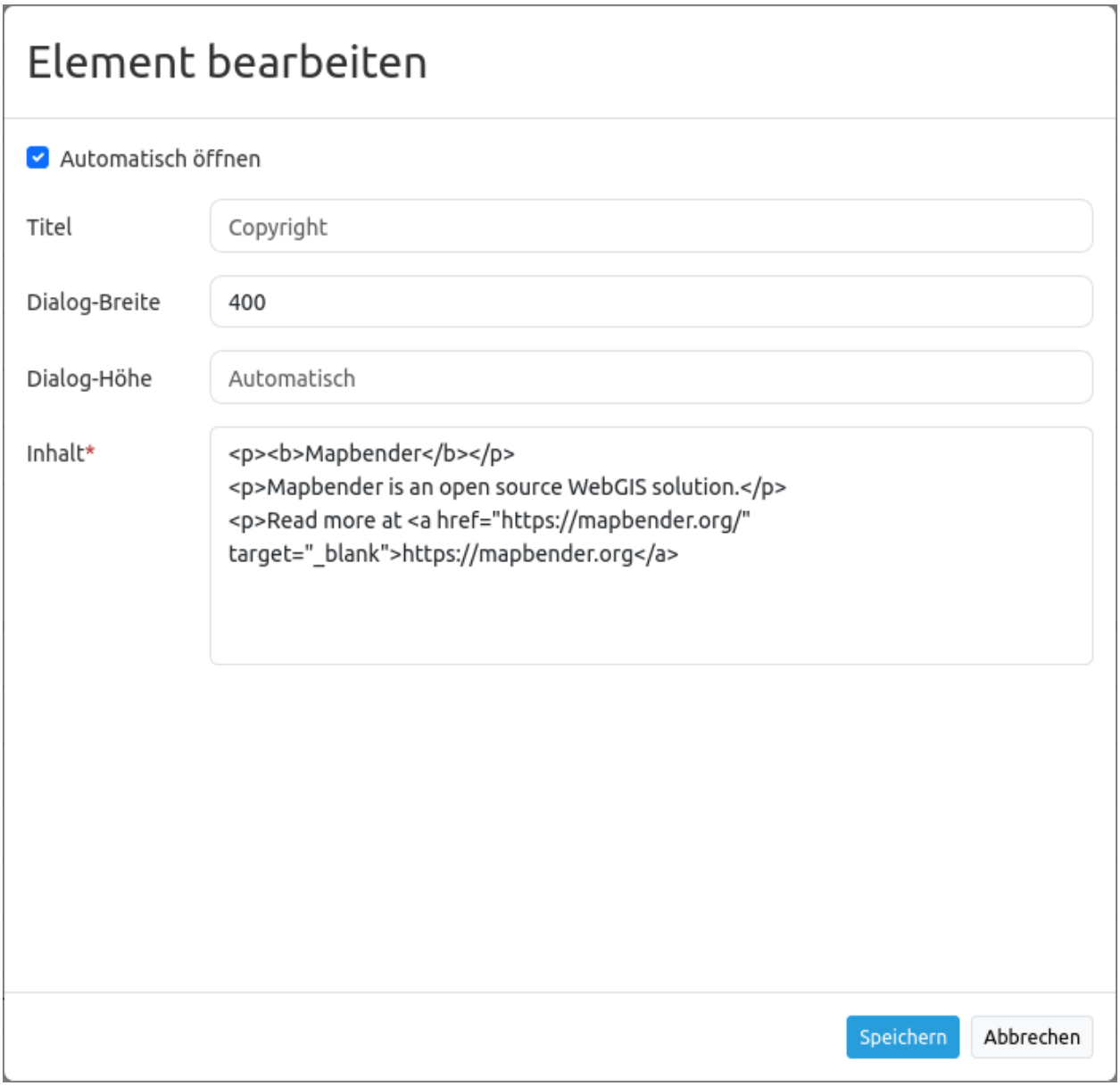
Tip: Nach dem Ändern des Parameters müssen Sie die Inhalte des Cache-Verzeichnisses (`var/cache/\*`) löschen.

3.6.4 Copyright

Dieses Element kann verwendet werden, um Text (auch HTML) in einem Dialog anzuzeigen. Der Dialog kann auf Wunsch beim Start automatisch erscheinen. Es können auch Links und Bilder integriert werden (siehe *HTML (Element)*).



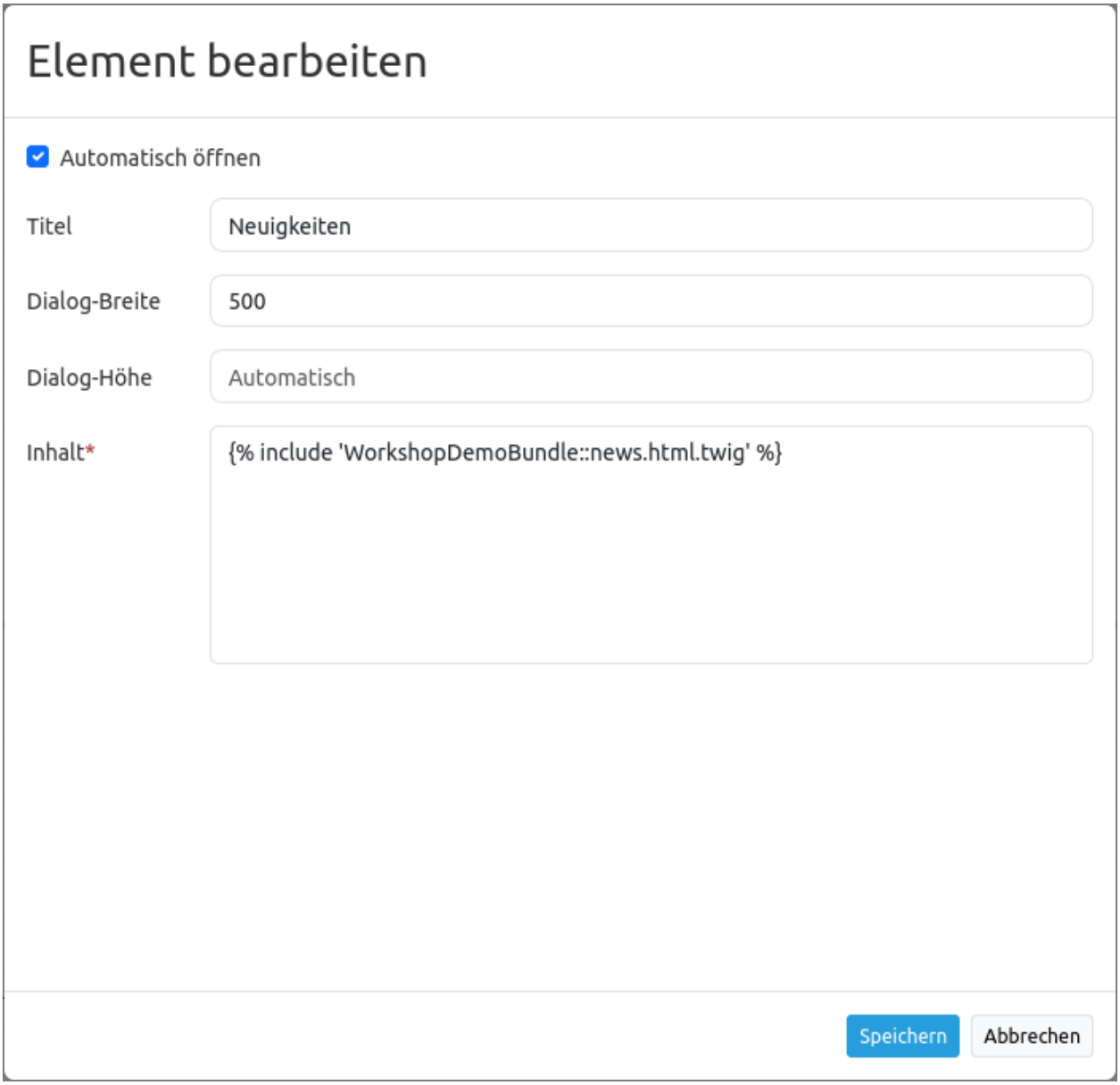
Konfiguration



- **Automatisch Öffnen:** Schaltet ein/aus, ob das Copyright Fenster beim Start der Anwendung automatisch geöffnet werden soll (Standard: aus).
- **Titel:** Titel des Elements. Der Titel wird neben dem Button angezeigt.
- **Dialog-Breite:** Breite des Popup Fensters (Standard: 300).
- **Dialog-Höhe:** Höhe des Popup Fensters (Standard: 170).
- **Inhalt:** Inhalt des Copyright Fensters. Dieser wird angezeigt, wenn das Element per Klick aktiviert wird (oder beim Start der Anwendung, wenn die “Automatisches Öffnen”-Option aktiviert wurde).

Verweis auf eine Twig-Datei

Im Content-Bereich kann auch auf eine Twig-Datei verwiesen werden. Bitte beachten Sie, dass die Twig-Datei valides HTML enthalten muss.



YAML-Definition

Diese Vorlage kann genutzt werden, um das Element in einer YAML-Anwendung einzubinden.

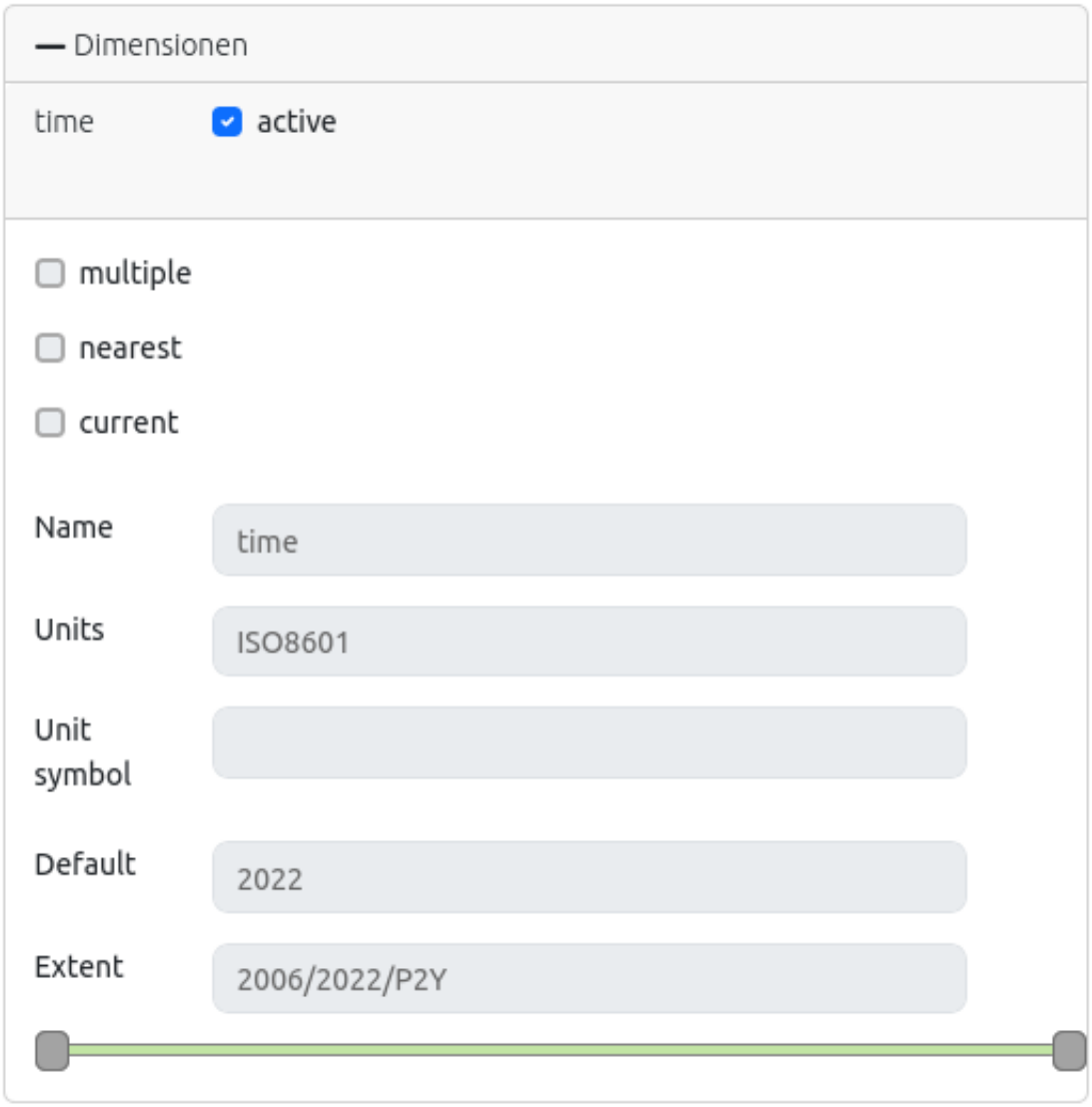
```yaml
class: Mapbender\CoreBundle\Element\Copyright
title: "Copyright"              # Titel des Elements
popupWidth: 300
popupHeight: 170
tooltip: "Copyright"            # Text des Tooltips
content: "Lorem ipsum"          # Erstellen Sie ihren Text für das Copyright
autoOpen: true                  # Automatisches Öffnen beim Start der Anwendung
```

3.6.5 Dimensions-Handler

Für die Anzeige von WMS-Diensten mit einer zeitlichen Dimension kann der Dimensions-Handler genutzt werden. WMS-Time Dienste müssen dafür zunächst als WMS-Datenquelle registriert werden.

WMS-T werden wie herkömmliche WMS in die Layersets eingefügt, zusätzlich muss jedoch der Time-Parameter aktiviert werden. Wird dieser nicht aktiviert, werden die Dimensionen des Dienstes außer Acht gelassen und beim Aufruf der Kartenebene der Standardwert genutzt.

Bei Diensten mit einer zeitlichen Dimension erscheint bei der Instanz die Schaltfläche **Dimensionen**. Durch einen Klick auf diese Schaltfläche werden die unterstützten Zeit-Parameter angezeigt. Die zeitliche Dimension kann dort per Checkbox-Klick aktiviert werden. Nach einem weiteren Klick in die Schaltfläche öffnet sich das Detailformular, in dem die Nutzung weiter definiert werden kann:



- **Abfrageart**: multiple, nearest, current
- **Name**: Wert TIME (name=time)
- **Units**: Abbildungsformat für zeitliche Dimensionsangaben (ISO 8601:2000)
- **Unit symbol**: Dienstabhängig variierendes Einheitensymbol (bspw. (y) für Jahre)
- **Default**: Standardzeitpunkt
- **Extent (extent slider)**: Unterstützter Zeitbereich

Nach der Dienste-Konfiguration unterstützt der Dimensions-Handler je nach Konfiguration die folgenden Zeitdimensionen:

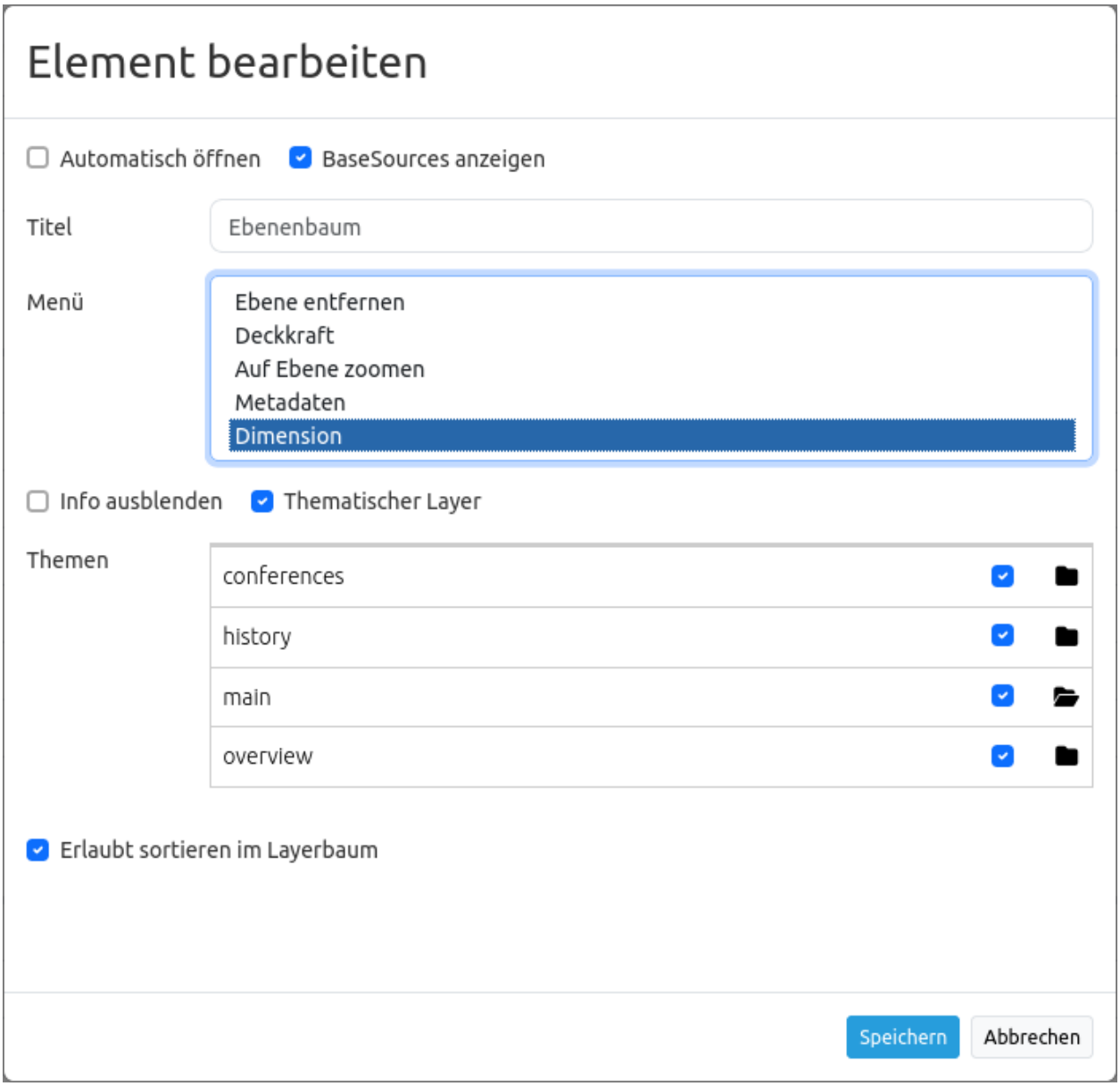
- einzelner Zeitpunkt
- Liste von Zeitpunkten
- Zeitintervall

Steuerung von Zeitvariablen

Es bestehen zwei Möglichkeiten, die Zeitangabe in der Anwendung zu steuern. Zum einen kann jeder Dienst, für den der Time-Parameter aktiviert ist, über das Kontextmenü des Layers im Layertree gesteuert werden. Zum anderen kann ein zentraler Schieberegler eingebunden werden, der in einem beliebigen Bereich der Anwendung angezeigt werden kann. Über den Schieberegler können mehrere Kartenebenen mit demselben Kartenbereich zusammengefasst und zentral gesteuert werden.

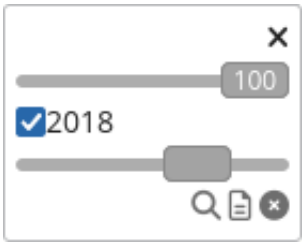
Schieberegler im Kontextmenü

Die Zeitachse kann über den Ebenenbaum als Option in das Kontextmenü des Layers im integriert werden. Dazu muss die Option **Dimension** im *Ebenenbaum (Layertree)* aktiviert werden.



Nach der Aktivierung in dem Ebenenbaum erscheint ein Schieberegler in dem Kontextmenü. Für die Nutzung der zeitlichen Anzeige muss das Element zuvor über die Checkbox aktviert werden. Danach kann

die Zeitachse mithilfe des Cursors verschoben werden.



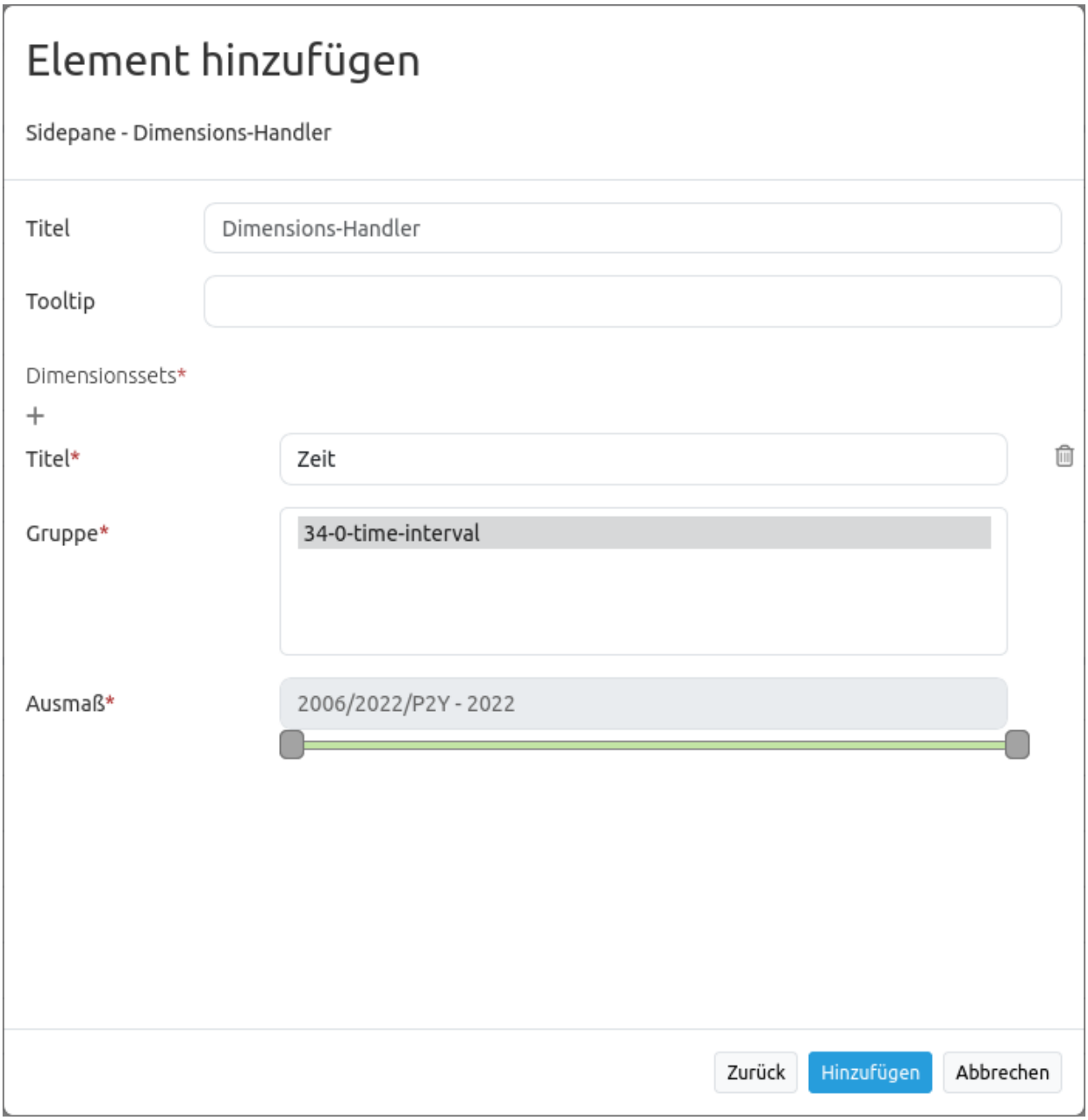
Schieberegler als Element

Die Kartenebenen können mithilfe des Dimensions-Handlers über einen zentralen Schieberegler gesteuert werden. Dieses Element kann direkt in die Sidepane oder über einen *Button* in die obere Werkzeugleiste und in die Fußzeile integriert werden. Die Konfiguration des Dimensions-Handlers erfolgt in drei Schritten:

- **Anlegen des Elements**: Zuerst muss das Element im *Backend* (Layout-Bereich) als neues Element in einen der Bereiche Sidepane, obere Werkzeugleiste oder Fußzeile hinzugefügt werden.
- **Definition eines Dimensionsets**: Anschließend erfolgt die Erstellung eines Dimensionsets über den „+"-Button. Hiernach muss ein Titel angegeben und eine Layerset-Instanz als Gruppe gewählt werden.
- **Definition des Schiebereglers**: Zuletzt muss der Bereich, der über den Schieberegler gesteuert werden sollen, festgelegt werden. Eine Mehrfachauswahl ist möglich, aber es können nur Instanzen kombiniert werden, die den gleichen zeitlichen Extent besitzen.

Wenn eine Instanz ausgewählt wurde, sind alle Instanzen, die nicht dieser Vorgabe entsprechen, nicht mehr auswählbar. Nach der Konfiguration kann das Dimensions-Handler Element gespeichert werden, um in der Anwendung zu erscheinen.

Konfiguration

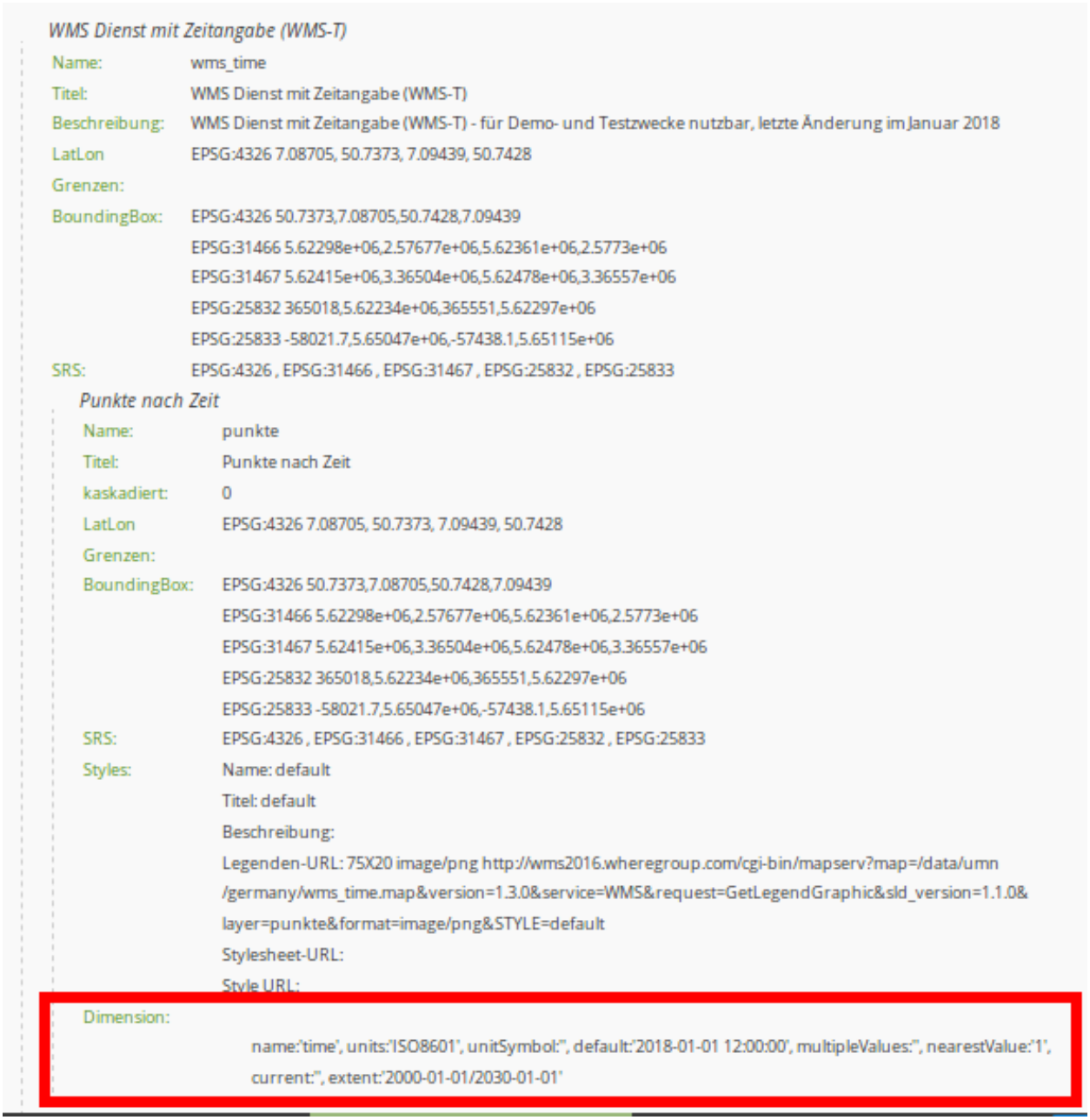


- **Title**: Titel des Elements
- **Tooltip**: Beschriftung des Elements bei Mouseover
- **Dimensionsset**: Gruppe für die zeitliche Dimensionsanzeige mit Titel (**Titel**), Layergruppe (**Gruppe**) und zeitlicher Dimensionsspanne (**Ausmaß**)

Im folgenden Abschnitt sind die YAML-Definitionen für das Element und die Einbindung in Dienst und Ebenenbaum aufgeführt.

Falls der Dienst über eine zeitliche Dimension verfügt, wird diese in den Layer-Metadaten angezeigt.

```
Dimension:
name:'time', units:'ISO8601', unitSymbol:'', default:'2018-01', multipleValues:'', nearestValue:'1',
↪current:'', extent:'2014-01/2018-01/P1M'
```



Elementkonfiguration

```
target: ~                           # ID des Kartenelements
dimensionsets:
    title: Zeit
    group:
        dimension:      # Dimensionsangabe wie in der Layerkonfiguration
            origextent: 2014-01/2018-01/P1M
            active: null
            type: interval
            name: time
            units: ISO8601
            unitSymbol: null
            default: 2017-11
            multipleValues: true
            nearestValue: true
            current: true
            extent: 2014-01/2017-11/P1M
```

Layerkonfiguration

```
dimension:                      # Dimensionsangabe in der Layerinstance
    -   name: time
        units: ISO8601
        unitSymbol: null
        default: 2018-01
        multipleValues: false
        nearestValue: true
        current: false
        extent: 2014-01/2018-01/P1M
```

Ebenenbaum

```
title: Layertree
class: Mapbender\CoreBundle\Element\Layertree
configuration:
    [...]
    menu:
        - [...]
        - dimension
```

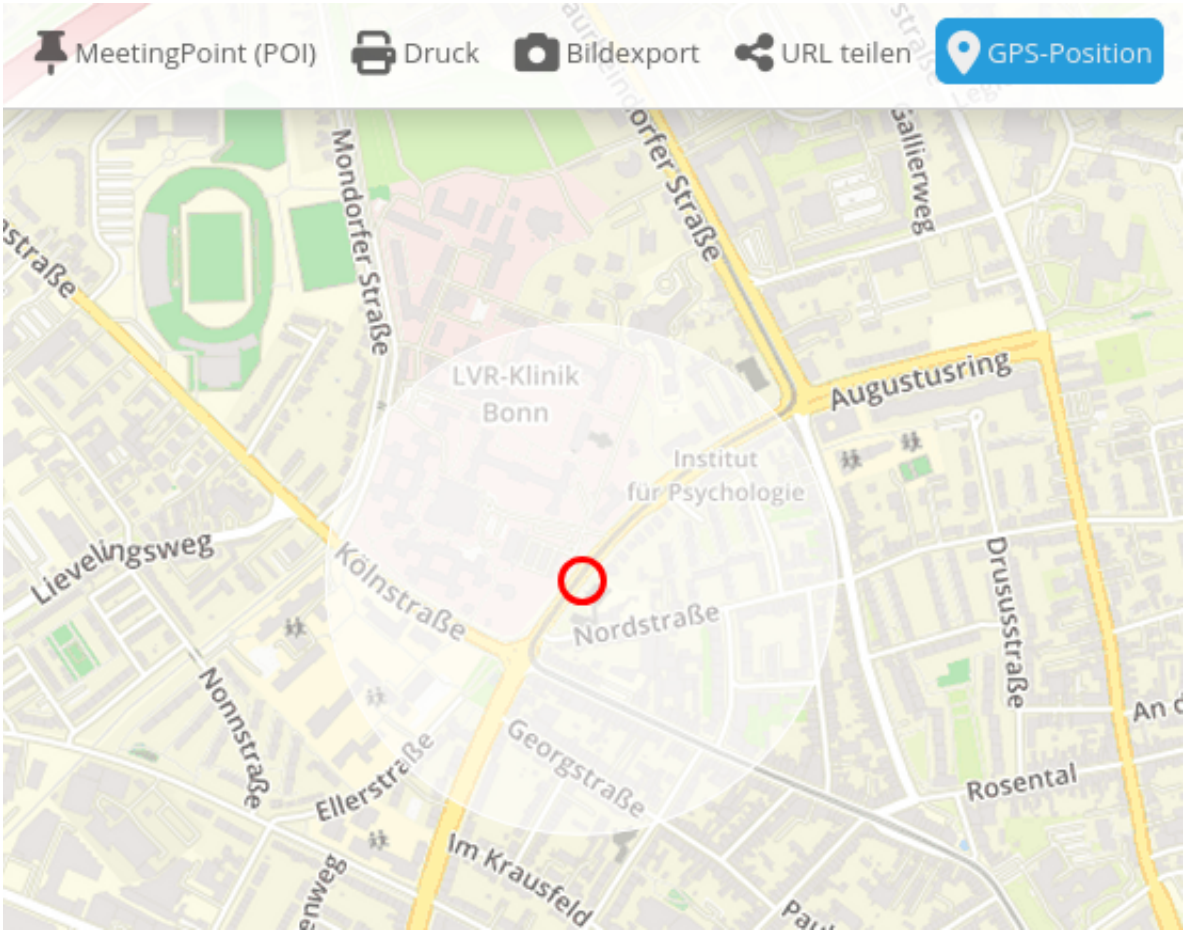
3.6.6 GPS-Position

Dieses Elememt stellt einen Button bereit, der zur aktuellen Position auf der Karte navigiert und ein markierendes Symbol anzeigt. Der Maßstab wird dabei nicht verändert, außer durch Aktivierung der Einstellung `zoom to accuracy (zoom to accuracy on first position)`.

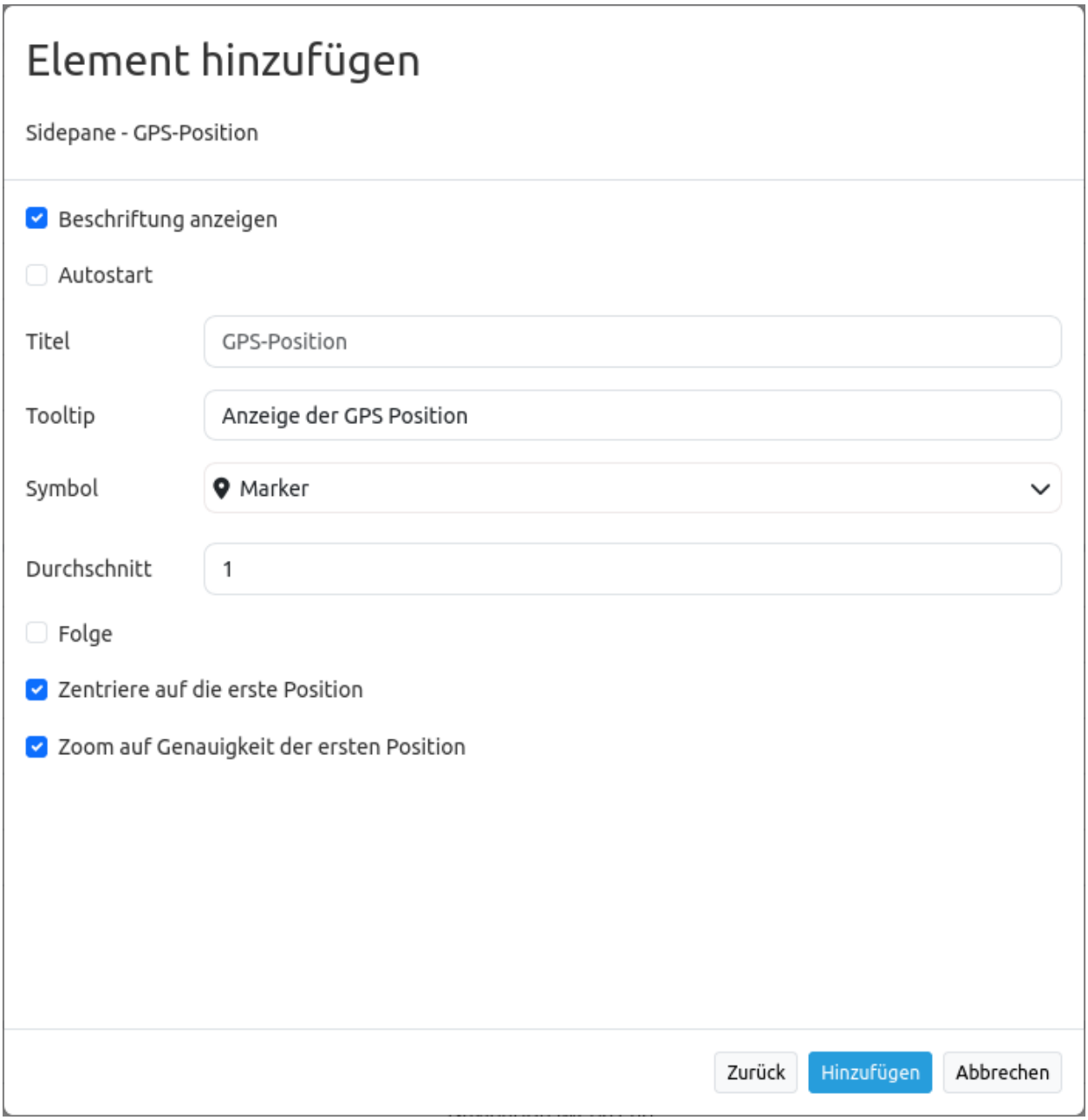
Die Funktion baut auf der Geolocation-API des W3C auf. Ob Ihr Browser diese Funktionalität unterstützt, erfahren Sie auf der Can I Use Seite. Die Funktion nutzt den `High Accuracy Parameter`, der die Standortbestimmung über GPS forciert. Falls das Gerät einen GPS-Empfänger hat und dieser aktiviert ist, ist die Positionsbestimmung also genauer. Ansonsten werden die WLAN Access-Points zur Positionsbestimmung herangezogen.

Der Mittelpunkt zeigt die wahrscheinliche Position des Gerätes an, der äußere Kreis die Genauigkeit der Positionsbestimmung, d.h. in welchem Bereich sich die ermittelte Position wahrscheinlich befindet.

Kompatibilität: Internet Explorer und MS Edge geben ohne GPS-Gerät am Rechner teilweise sehr ungenaue Informationen heraus. Das Verhalten ist auch mit anderen Anwendungen zu beobachten.



Konfiguration



- **Beschriftung anzeigen:** Schaltet die Beschriftung des Buttons an/aus (Standard: an).
- **Autostart** Startet Element beim Öffnen der Anwendung (Standard: aus).
- **Titel:** Titel des Elements. Dieser wird in der Layouts Liste angezeigt und ermöglicht, mehrere Button-Elemente voneinander zu unterscheiden. Der Titel wird außerdem neben dem Button angezeigt, wenn "Beschriftung anzeigen" aktiviert ist.
- **Tooltip:** Text, der angezeigt wird, wenn der Mauszeiger eine längere Zeit über dem Element verweilt.
- **Symbol:** Symbol des Buttons, basierend auf einer CSS Klasse.

- **Durchschnitt:** Berechnet den Mittelwert der unter Average angegebenen letzten empfangenen GPS Koordinaten (Standard: 1).
- **Folge:** Positioniert die Karte bei jeder empfangenen GPS Koordinate neu (Standard: false).
- **Zentriere auf die erste Position:** Zentriert die Karte auf die erstermittelte Position (Standard: true).
- **Zoom auf Genauigkeit der ersten Position:** Zoomt auf die ermittelte Koordinate nach Messgenauigkeit bei erster ermittelten Position (Standard: true).

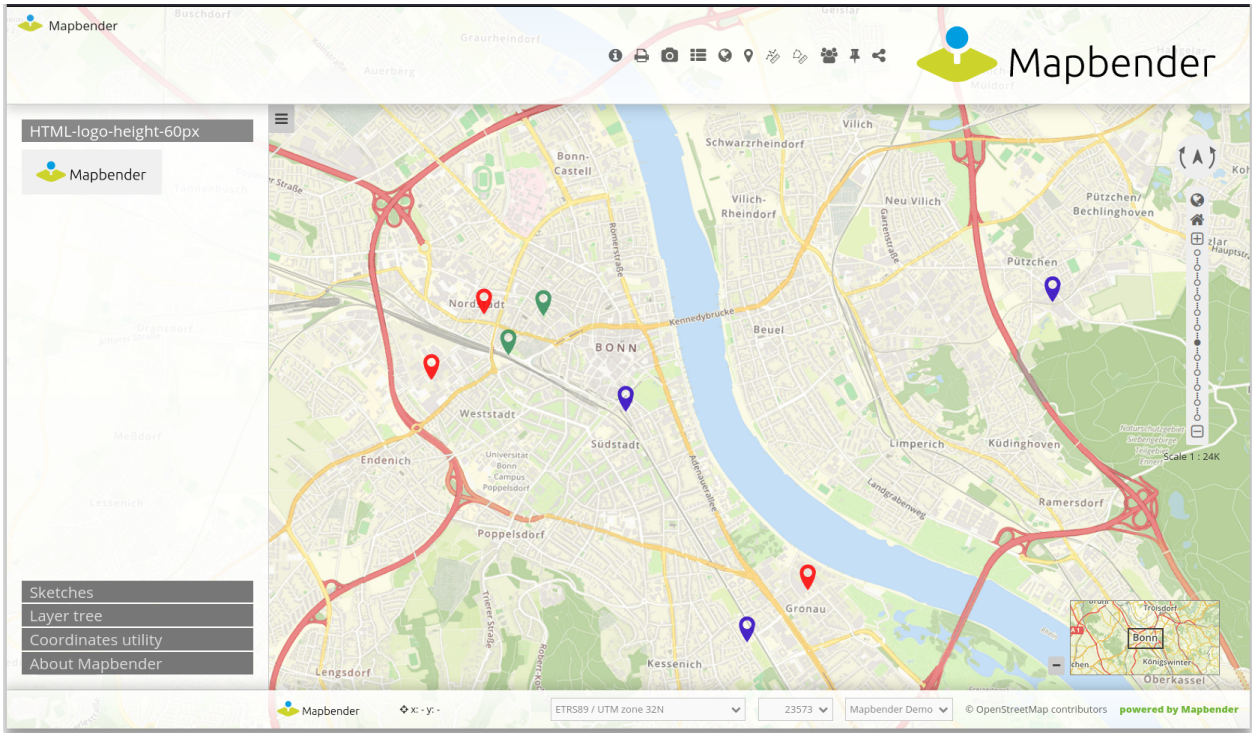
YAML-Definition

Diese Vorlage kann genutzt werden, um das Element in einer YAML-Anwendung einzubinden.

```
class: Mapbender\CoreBundle\Element\GpsPosition
label: true                         # false/true, um den Button zu beschriften (Standard: true)
autoStart: false                    # true, wenn diese Funktion beim Start der Anwendung geöffnet
↪werden soll (Standard: false).
title: GPS-Position                 # Titel des Buttons
tooltip: GPS-Position               # Text des Tooltips
icon: iconGpsTarget                 # Symbol für den Button
target: map                         # ID des Kartenelements
average: 1                          # berechnet den Mittelwert der unter average angegebenen letzten
↪empfangenen GPS Koordinaten (Standard: 1).
refreshinterval: 5000               # Aktualisierungsintervall in ms (Standard: 5000 ms).
follow: true                        # True positioniert die Karte bei jeder empfangenen GPS Koordinate
↪neu (Standard: false). Sollte nur mit WMS Diensten im gekachelten Modus verwendet werden, da sonst
↪bei jeder Neupositionierung ein neuer Kartenrequest geschickt wird
centerOnFirstPosition: true         # Zentriert die Karte auf die erstermittelte Position
zoomToAccuracy: false               # Zoomt auf die ermittelte Koordinate nach Messgenauigkeit
zoomToAccuracyOnFirstPosition: true # Zoomt auf die ermittelte Koordinate nach Messgenauigkeit bei
↪erster ermittelten Position
```

3.6.7 HTML (Element)

Dieses Element ermöglicht die Definition von HTML an beliebigen Stellen in der Anwendung. Dadurch können zum Beispiel Bilder, Links oder Texte eingebunden werden. Die folgende Abbildung zeigt die Integrierung des Mapbender-Logos in Fußzeile, oberer Werkzeugleiste und Sidepane.

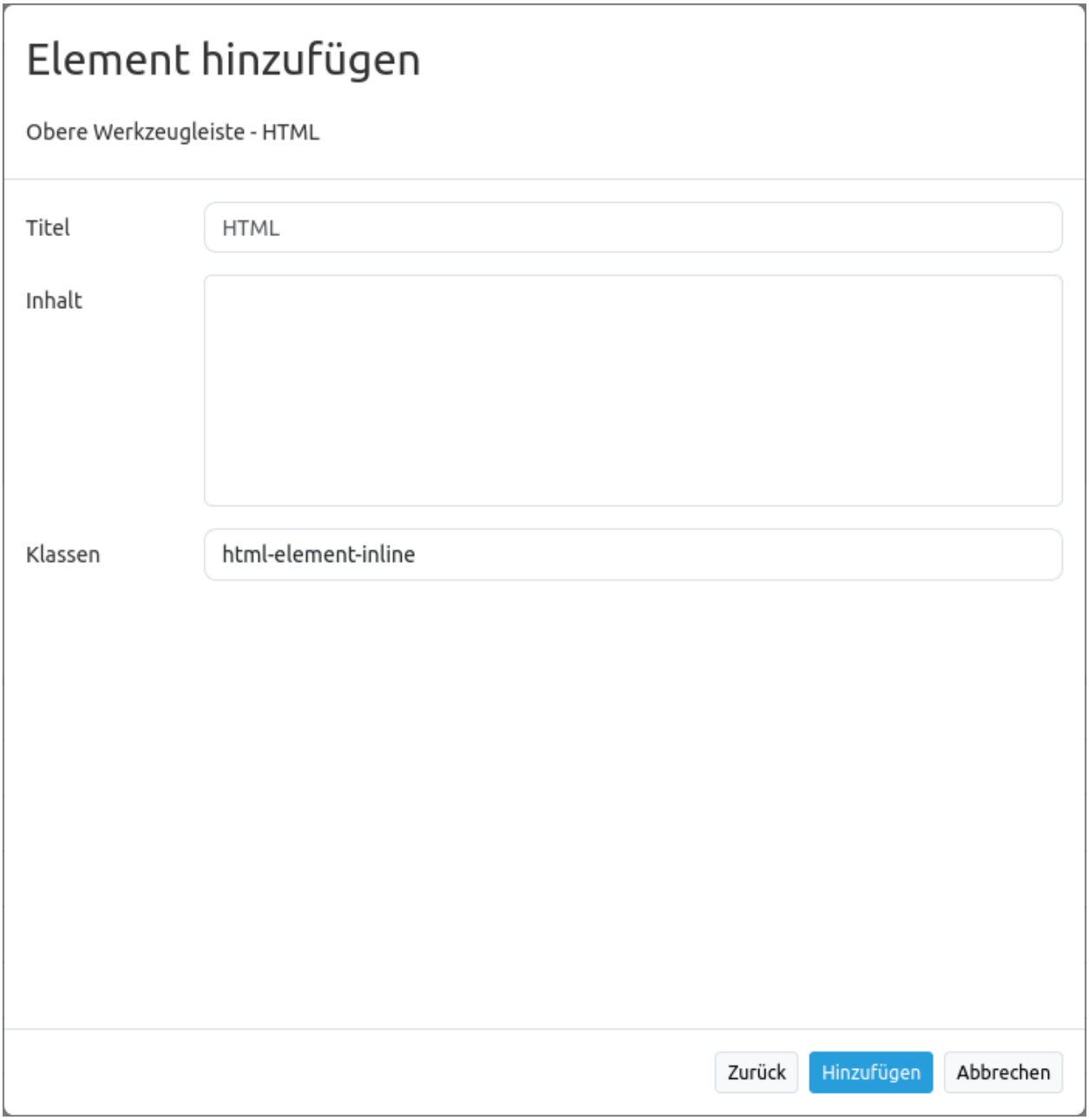


Es stehen grundsätzlich die folgenden Variablen im HTML-Element zur Verfügung:

- “application” (Entity Application)
- “entity” (Entity HTML-Element)

Konfiguration

Das Element wird im *Backend* in die Anwendung eingebunden.



- **Titel:** Titel des HTML-Elements. Dieser wird in der Layouts-Liste angezeigt und ermöglicht, mehrere HTML-Elemente voneinander zu unterscheiden.
- **Inhalt:** Inhalt des HTML-Elements, z.B. Bild oder Link. Der Content kann die Variablen “application” und “entity” beinhalten.
- **Klassen:** Alternativ kann auch auf eine CSS-Klasse für das HTML-Element verwiesen werden. Die im CSS-Editor definiert werden kann.

Konfigurationsbeispiele

Bild einbinden:

Das Mapbender Logo wurde im Beispiel als Bild in die Seitenleiste eingebunden. Es kann eine definierte Höhe angegeben werden. Es kann eine Hintergrundfarbe, ein bestimmter Seitenrandabstand und die Transparenz gesetzt werden. Diese Eigenschaften werden im Styling-Bereich definiert.

```
<img src='https://mapbender.org/fileadmin/mapbender/resources/images/logos/Mapbender-Logo.svg'
height='60px' style='background-color:rgb(240, 240, 240, 0.9); padding:10px'>
```

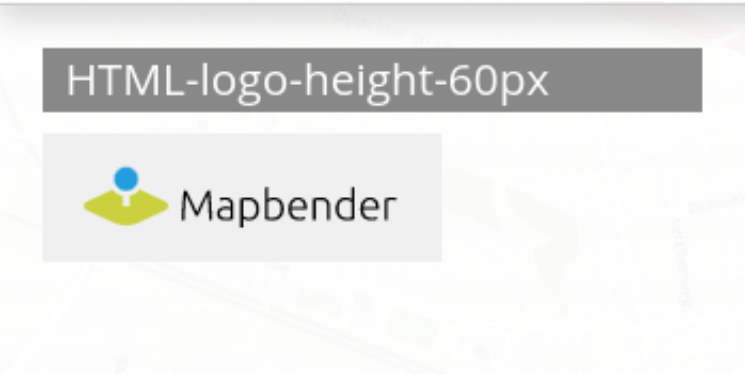


Bild mit hinterlegtem Link einbinden

Das zuvor eingebundene Logo kann um eine Linkverknüpfung erweitert werden. Der Link verweist auf die Mapbender-Webseite. Die Information `target='_blank'` sorgt dafür, dass sich der Link beim Klicken in einem neuen Tab öffnet.

```
<a href='https://mapbender.org' target='_blank'>
<img src='https://mapbender.org/fileadmin/mapbender/resources/images/logos/Mapbender-Logo.svg'
height='60px' style='background-color:rgb(240, 240, 240, 0.9); padding:10px'> </a>
```

Variablen & HTML-Element

Es besteht die Möglichkeit, Variablen in Anwendungen zu verwenden.

- Variable: **“application.title”**

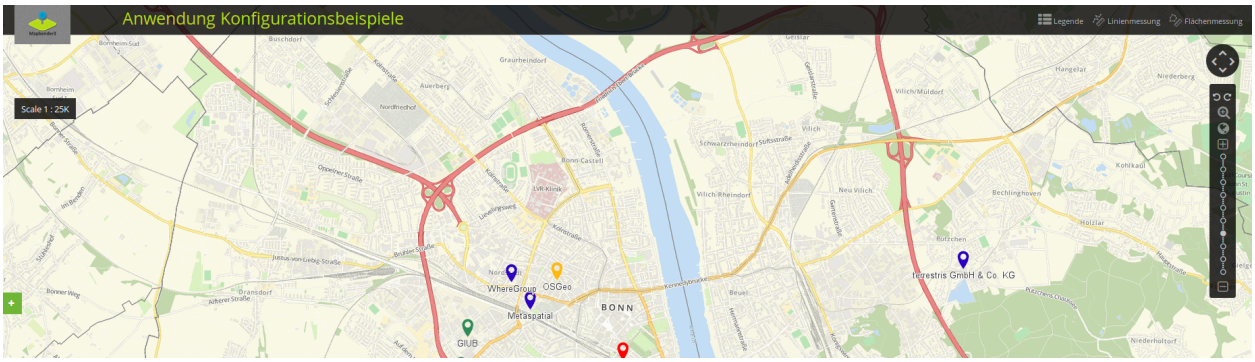
Mit dieser Variable kann der Titel einer Anwendung eingebunden werden. Im folgenden Beispiel lautet dieser: “Konfigurationsbeispiele”.

Der HTML-Code kann wie folgt aussehen:

```
<b><span style="font-size:25px;color:#b6dd18;margin-right:50vw"> Anwendung {{ application.title }} </
↪span></b>
```

Die Variable für den Anwendungstitel wird durch `{{ application.title }}` eingebunden. Der Begriff “Anwendung” ist ein Textzusatz und erscheint unabhängig vom Titel. Durch den Style-Block wurden Schriftgröße, Schriftbreite, Schriftfarbe sowie Position des Titels angepasst.

Für das Anwendungsbeispiel sieht das Ergebnis des HTML-Elements wie folgt aus:

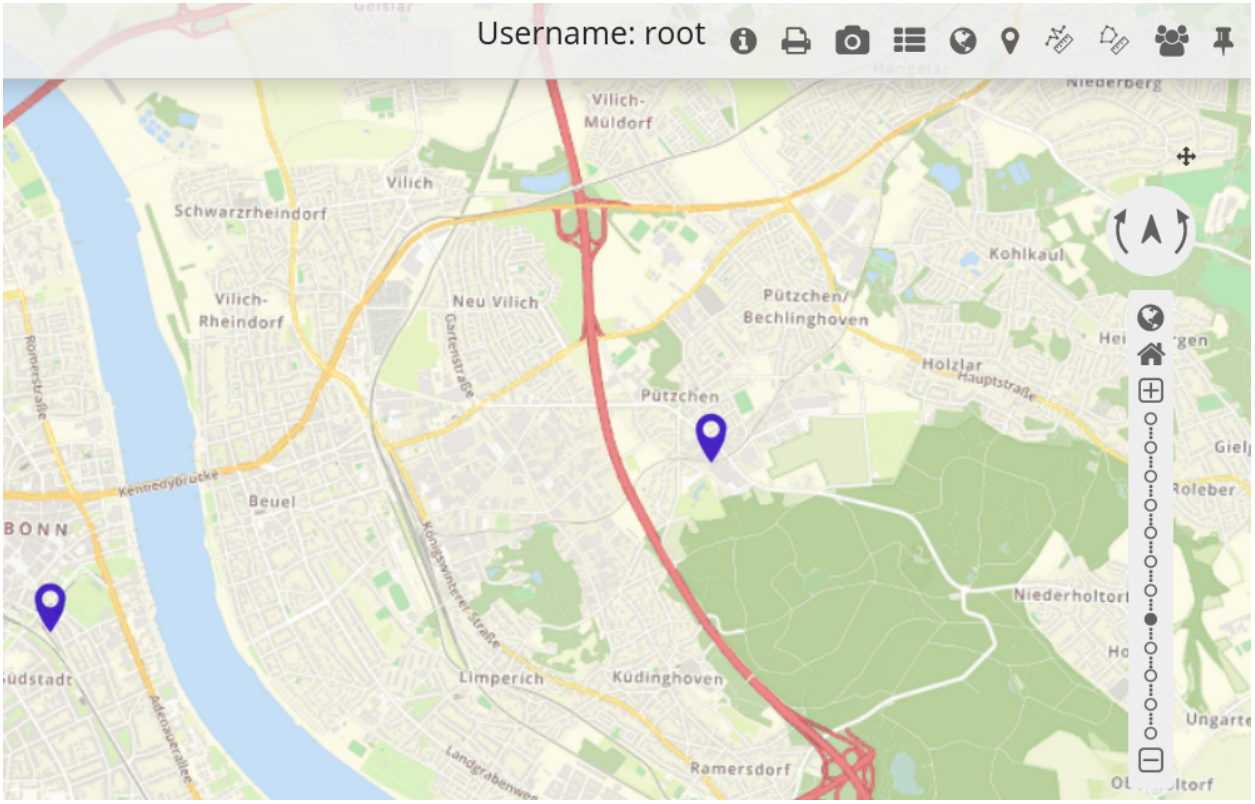


- Variable: **app.user.username**

Diese Variable gibt den Namen des aktiven Nutzers im HTML-Element aus:

```
<p>Username: {{ app.user.username }}</p>
```

Im Anwendungsbeispiel wird der Nutzername in der oberen Werkzeugleiste angezeigt:

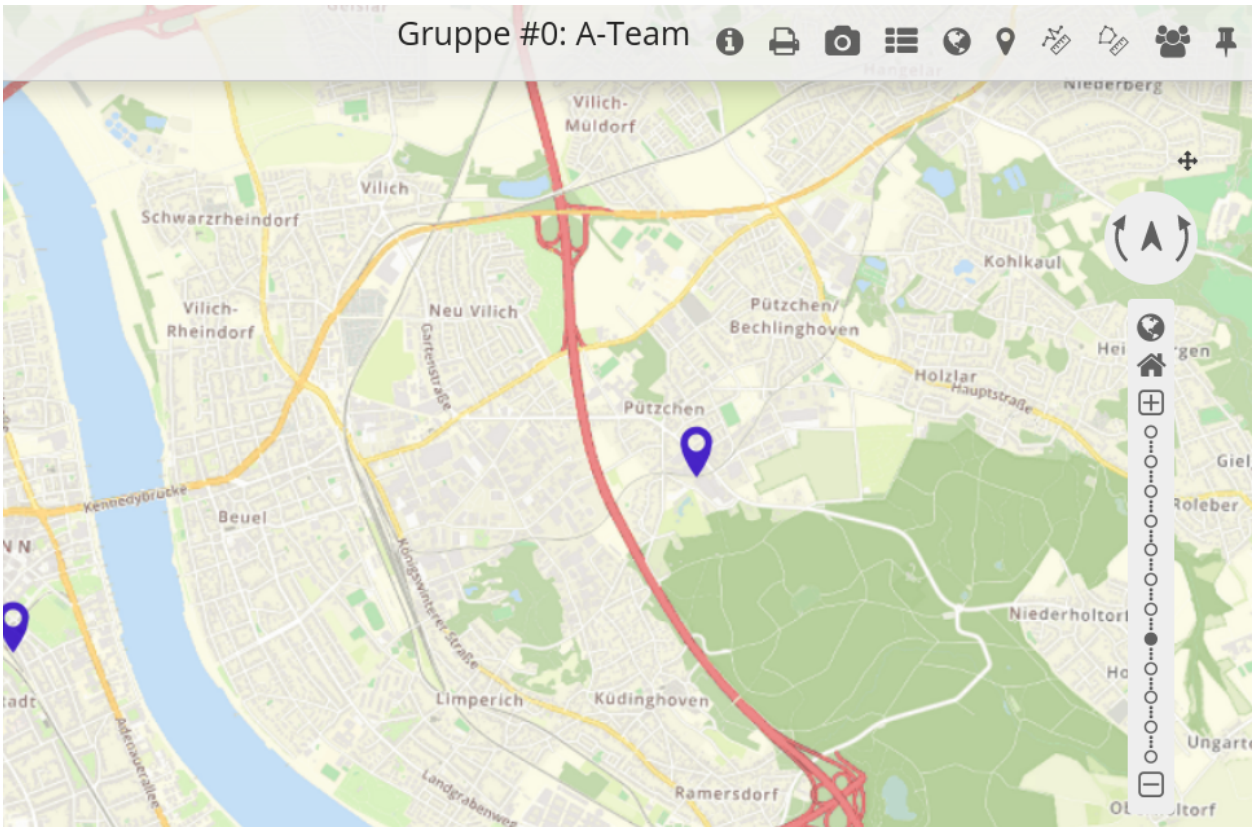


- Variable: **group.title**

Die Gruppe des Benutzers kann nicht in einem einzelnen Ausdruck wiedergegeben werden, da Twig den Map-Filter erst in höheren Versionen unterstützt. Um Gruppentitel dennoch ausgeben zu können, wird eine Schleife im HTML-Element eingebunden:

```
{% for index, group in app.user.groups %}
    <p>Gruppe #{{ index }}: {{ group.title }}</p>
{% endfor %}
```

Im Anwendungsbeispiel werden Index und Name der Gruppe in der oberen Werkzeugleiste ausgegeben:



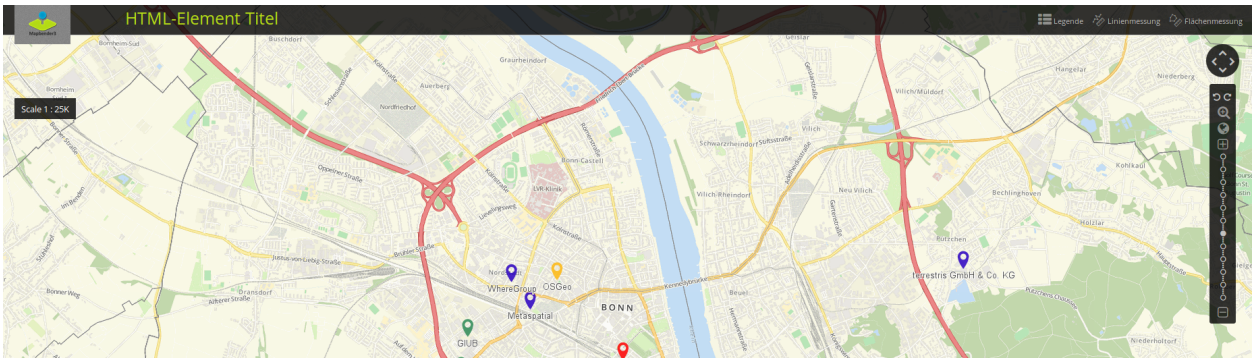
- Variable: **“entity”**

Die Variable `{ entity }` gibt die ID und `{ entity.title }` den Namen des HTML-Elements wieder.

Im Folgenden wurde die Variable `{ entity.title }` mit dem Textzusatz “HTML-Element” eingebunden. Die Styling-Parameter entsprechen der Konfiguration aus dem Beispiel *application.title*. Für *entity.title* wurden lediglich Textzusatz, Variable und Position (margin-right) angepasst.

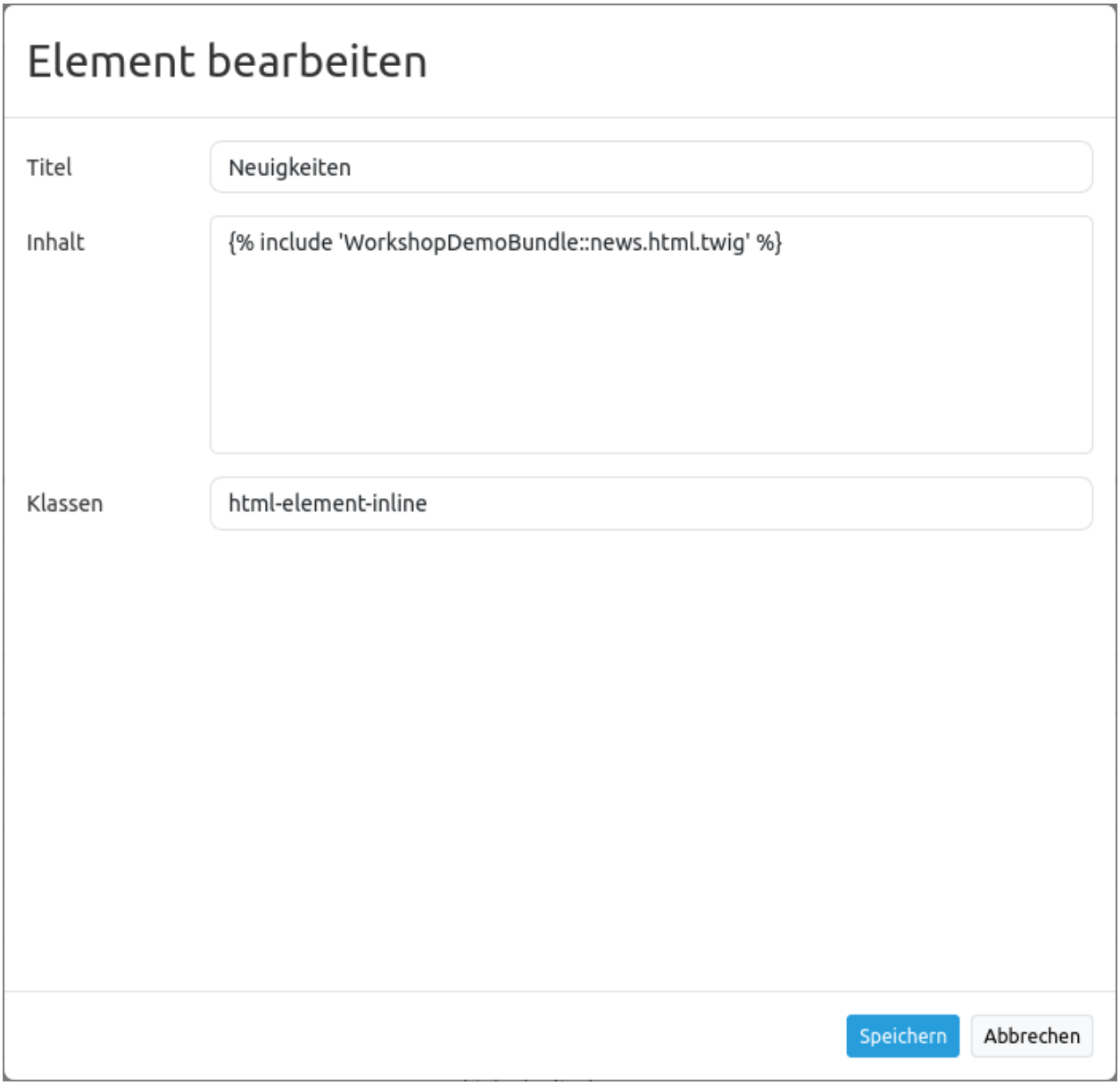
```
<b><span style=“font-size:25px;color:#b6dd18;margin-right:60vw“> HTML-Element {{ entity.title }} </span>
↪</b>
```

Diese Variablen sehen in der Anwendung wie folgt aus:



Verweis auf eine Twig-Datei

Im Bereich *Inhalt* kann auch auf eine Twig-Datei verwiesen werden. Bitte beachten Sie, dass die Twig-Datei valides HTML enthalten muss.



YAML-Definition

Diese Vorlage kann genutzt werden, um das Element in einer YAML-Anwendung einzubinden.

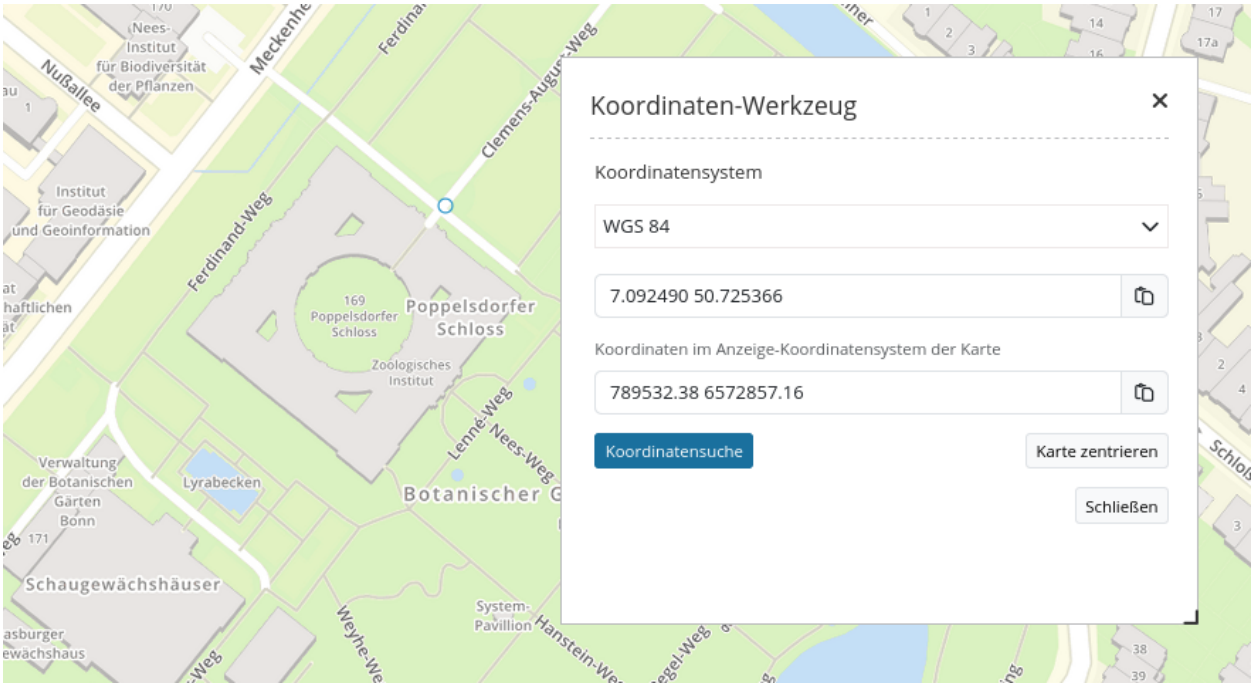
```
title: 'HTML-Element'
class: Mapbender\CoreBundle\Element\HTMLElement
content: <p>Hello, World!</p><p>Application: {{ application.title |trans }}</p> #content kann die
↪Variablen "application" und "entity" beinhalten.
classes: my-special-css-class
```

3.6.8 Koordinaten Utility

Koordinaten Utility erlaubt zwei verschiedene Koordinatenoperationen:

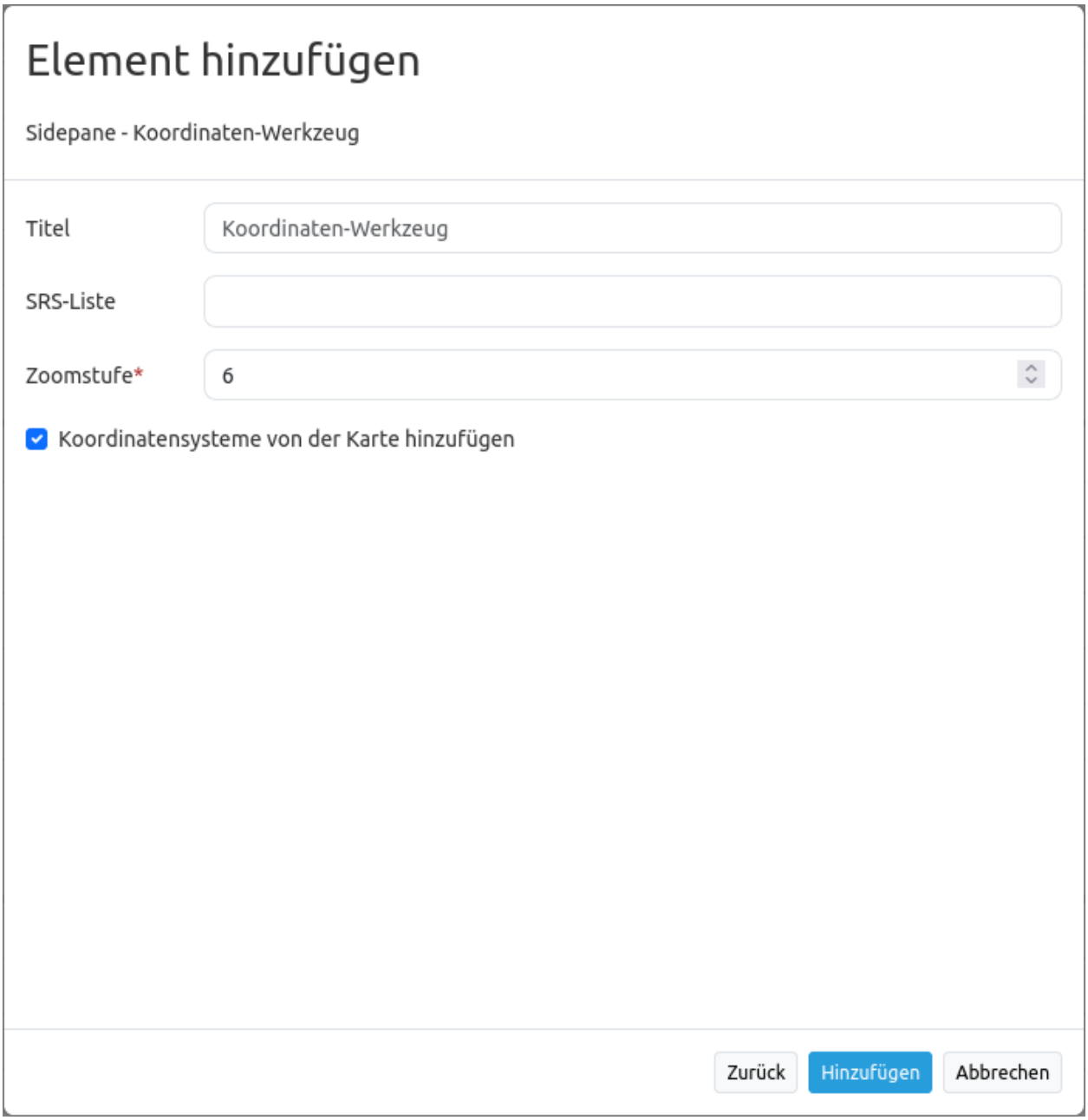
1. Es kann zu einem Punkt mithilfe des zugehörigen eingegebenen Koordinatenpunkts gesprungen werden.
2. Es kann der Klickpunkt aus der Karte anzeigt werden.

Beide Funktionen werden über eine Oberfläche realisiert.



In beiden Fällen ist zusätzlich eine dynamische Koordinatentransformation möglich, so dass auch Punkte anderer Koordinatensysteme eingegeben werden können.

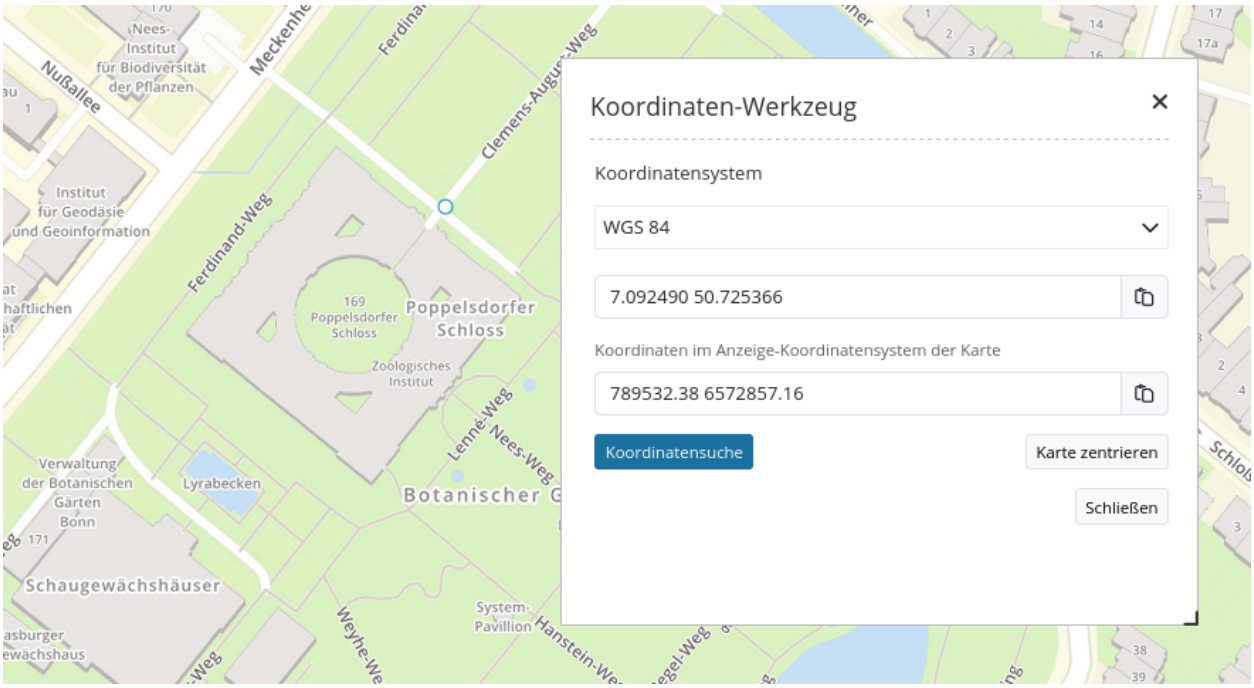
Konfiguration



- **Titel:** Titel des Elements
- **SRS-Liste:** Hier können weitere SRS angegeben werden, in denen das Tool transformieren können soll. Das Feld kann leer gelassen werden.
- **Zoomstufe** Zoom-Level der Karte (Standard: 6)
- **Koordinatensysteme von der Karte hinzufügen:** Ist die Checkbox aktiviert, so werden die von der *Karte (Map)* angegebenen Koordinatensysteme automatisch vom Tool unterstützt. Diese werden in der Karte angegeben und auch im *SRS Auswahl (SRS Selector)* unterstützt.

Sofern das Koordinaten Utility im Kartenbereich eingebunden werden soll, muss ein damit verknüpfter *Button* entsprechend platziert werden.

Benutzung



Koordinate abgreifen

- Ist das Koordinaten Utility als Dialog geöffnet, reagiert die Karte auf einen Klick. Klicken Sie in die Karte und die Klickkoordinate wird in dem Dialog angezeigt.
- Ändern Sie das Koordinatensystem im Dialog mithilfe der Drop-Down-Liste. Die Klick-Koordinate wird in diesem angegebenen Koordinatensystem angezeigt.
- Die untere Zeile zeigt daher die Klickkoordinate noch in dem Original-Koordinatensystem der Karte.
- Die Schaltfläche hinter jeder Eingabezeile erlauben es, den Koordinatenwert direkt in die Zwischenablage zu kopieren.
- Wenn das Koordinaten Utility über die Sidepane eingebunden ist, kann über den Button **Koordinatensuche** zwischen einem Navigations- und einem Kooordinaten-Cursor gewechselt werden.

Zur Koordinate zoomen

- In dem Textfeld können eigene Koordinaten angegeben werden. Diese müssen in dem Koordinatensystem eingegeben werden, welches in der Drop-Down-Liste darüber festgelegt wurde.
- Über einen Klick auf die Schaltfläche **Karte zentrieren** wird zu dem Punkt gezoomt und dieser mit einem orangenen Symbol angezeigt.

YAML-Definition

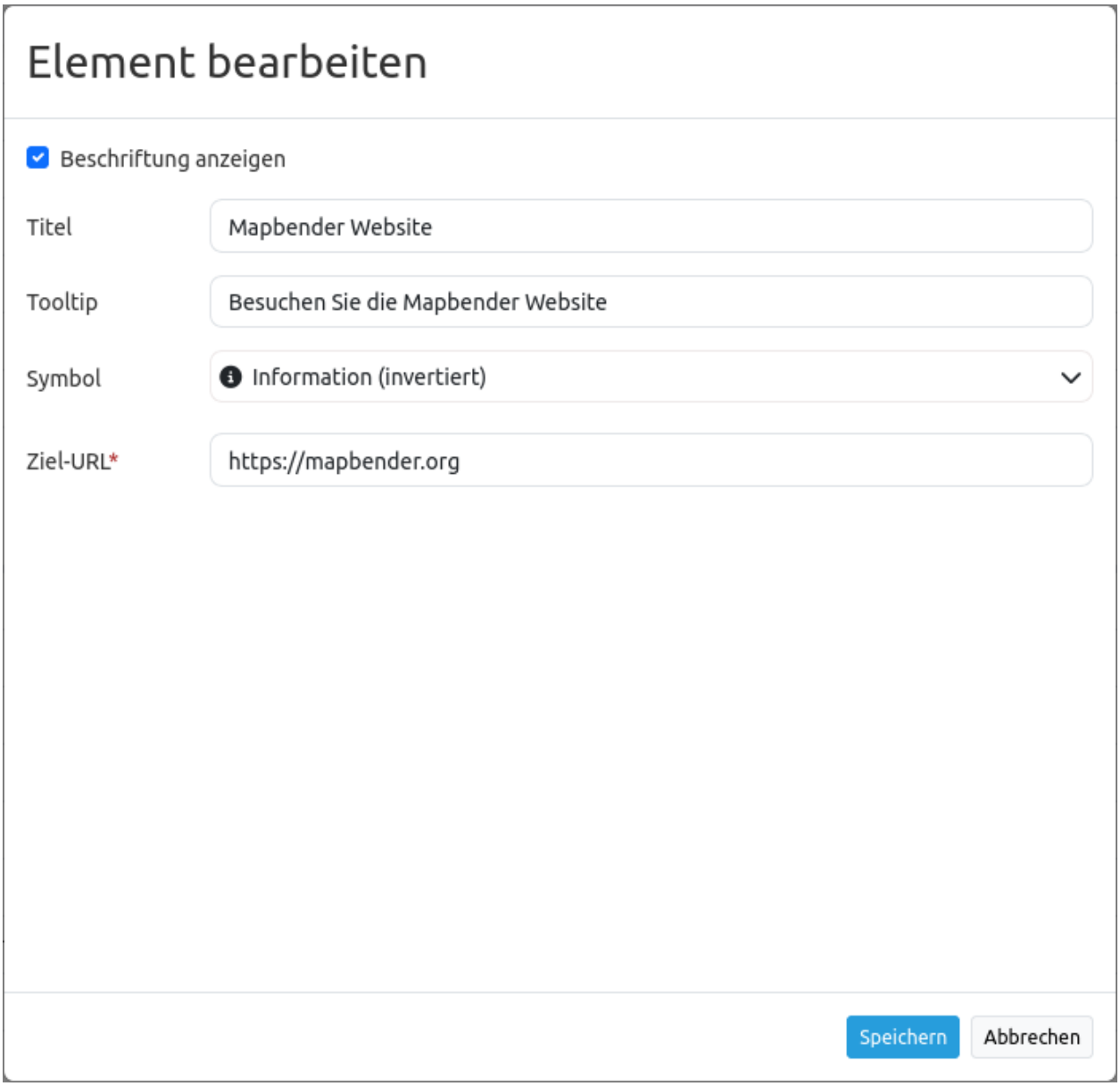
Diese Vorlage kann genutzt werden, um das Element in einer YAML-Anwendung einzubinden.

```
coordinatesutility:
    title: 'Koordinaten Utility'
    class: Mapbender\CoordinatesUtilityBundle\Element\CoordinatesUtility
    type: element
    target: map
    srsList:
        -
            name: 'EPSG:31466'
            title: '31466'
        -
            name: 'EPSG:31468'
            title: '31468'
        -
            name: 'EPSG:25833'
            title: '25832'
        -
            name: 'EPSG:4326'
            title: '4326'
            addMapSrsList: true
```

3.6.9 Link

Dieses Element stellt ein Button-Modul bereit, über das eine Webseite oder ein Skript verlinkt werden kann.

Konfiguration



- **Beschriftung anzeigen:** Schaltet die Beschriftung des Buttons an/aus (Standard: an).
- **Titel:** Titel des Elements. Dieser wird in der Layouts Liste angezeigt und ermöglicht, mehrere Button-Elemente voneinander zu unterscheiden. Der Titel wird außerdem neben dem Button angezeigt, falls die Checkbox "Beschriftung anzeigen" aktiviert ist.
- **Tooltip:** Text, der angezeigt wird, wenn der Mauszeiger eine längere Zeit über dem Element verweilt.
- **Symbol:** Symbol des Buttons, basierend auf einer CSS-Klasse.
- **Ziel-URL:** Angabe der Ziel-URL, auf die der Button verweist.

Hint: Es ist auch möglich, ein Icon-Set zu deaktivieren und/oder andere Icons zu verwenden. Weitere

Informationen finden Sie unter *Icons anpassen*.

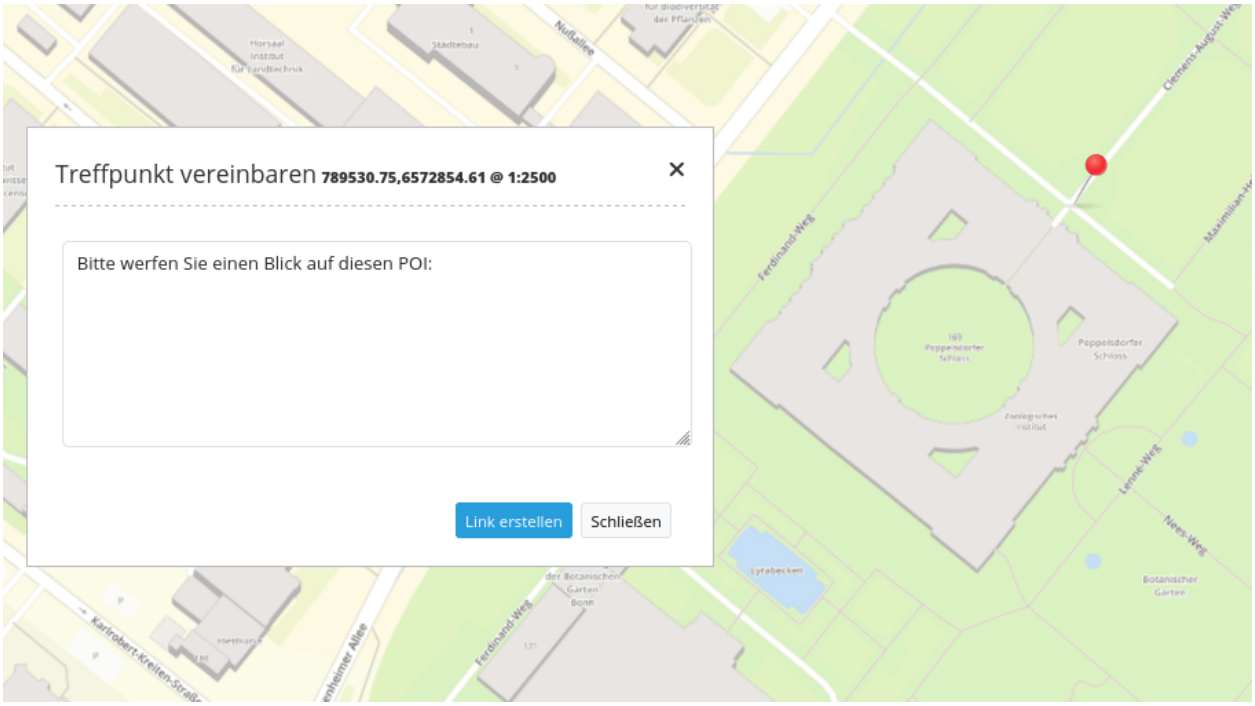
YAML-Definition

Diese Vorlage kann genutzt werden, um das Element in einer YAML-Anwendung einzubinden.

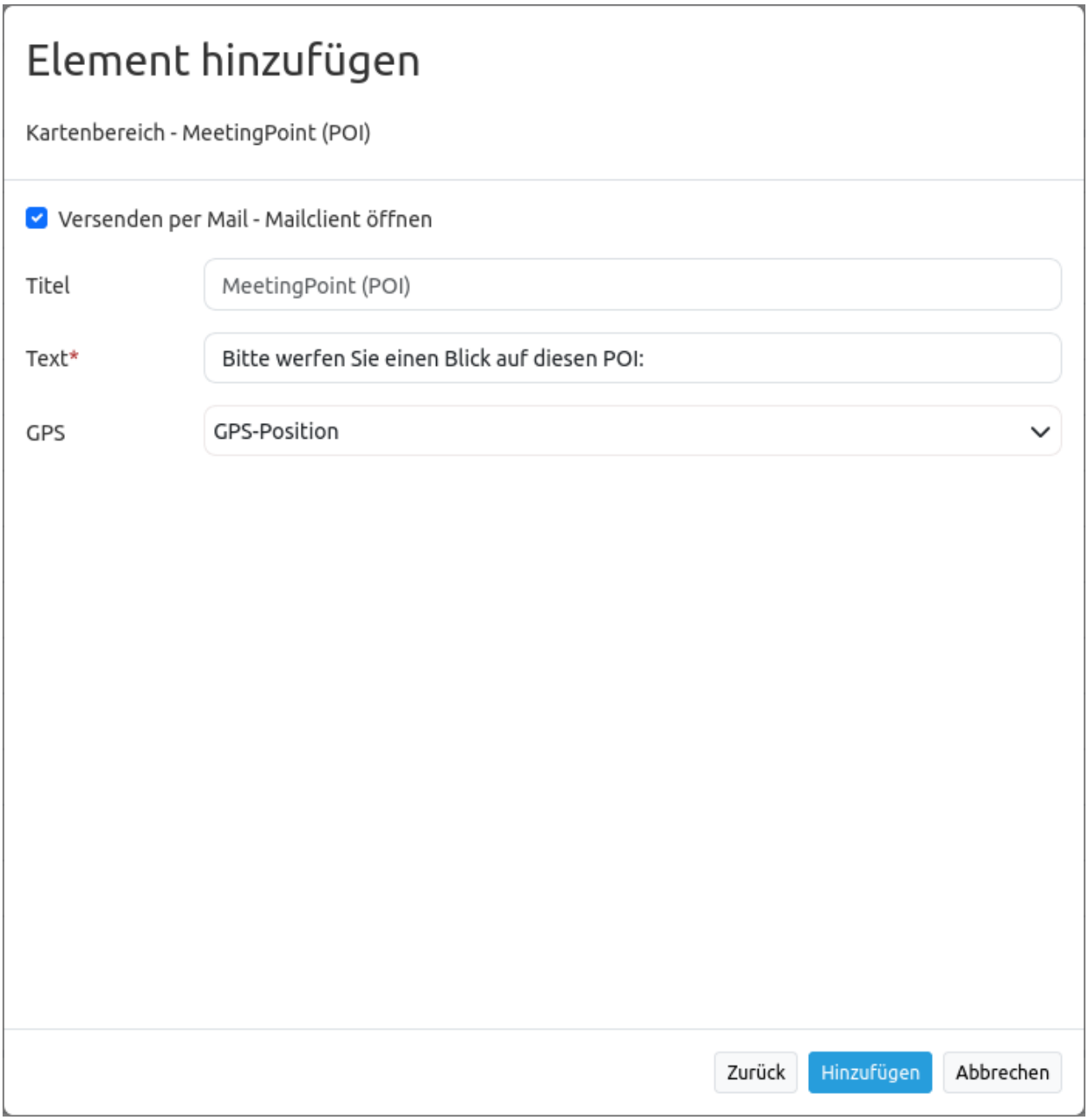
```
title: Link                                 # Titel
class: Mapbender\CoreBundle\Element\Button
tooltip: Besuchen Sie diese Webseite        # Text des Tooltips
icon: iconInfoActive                        # Symbol verwendete CSS Klasse
label: true                                 # false/true, um den Button zu beschriften (Standard: true).
click: https://mapbender.org                # bezieht sich auf eine Webseite oder ein Skript.
```

3.6.10 MeetingPoint (POI)

Generiert POI-URLs (Treffpunkt-URLS), verwendbar für das Verschicken eines Koordinatenpunkts per E-Mail. Dabei wird der übermittelte Punkt in das Koordinatensystem der Karte projeziert und angezeigt.



Konfiguration



- **Versenden per Mail - Mailclient öffnen:** Der POI wird per Mail verschickt.
- **Titel:** Titel des Elements. Dieser wird in der Layouts Liste angezeigt und ermöglicht, mehrere Button-Elemente voneinander zu unterscheiden. Der Titel wird außerdem neben dem Button angezeigt, wenn “Beschriftung anzeigen” aktiviert ist.
- **Text:** Dialogtext der nach dem Setzen eines POI angezeigt wird.
- **GPS:** Erlaubt eine GPS Positionierung.

YAML-Definition

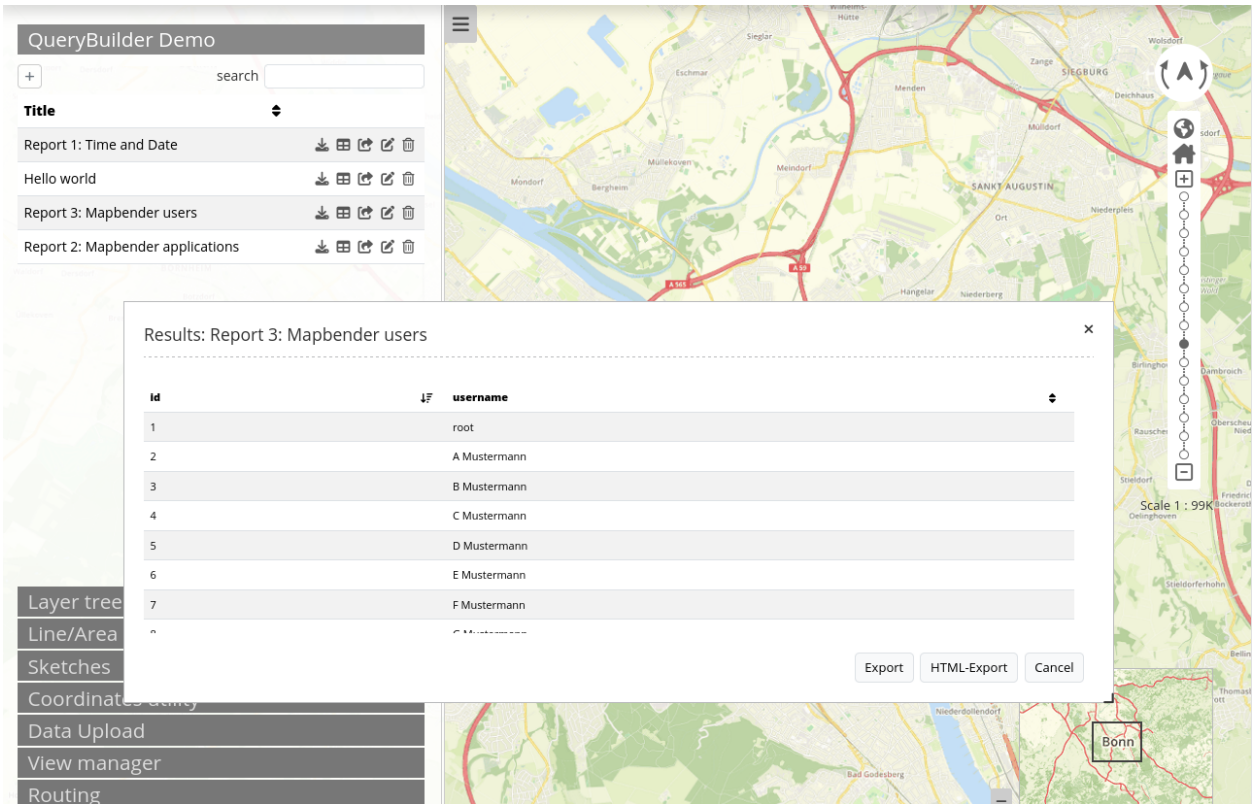
Diese Vorlage kann genutzt werden, um das Element in einer YAML-Anwendung einzubinden.

```
target: map                              # only mapp-element is possible
body: 'Setzen Sie per Klick einen Treffpunkt'  # definieren Sie einen Text für den Dialog
```

3.6.11 Query Builder

Der QueryBuilder ist nicht im Standardumfang von Mapbender enthalten, kann aber ergänzt werden. Der QueryBuilder stellt ein Element bereit, über das Auswertungen aus der Datenbank angezeigt werden können. Sie können SQL-Abfragen definieren und mit einem Titel vergeben. Diese werden in der Anwendungen als Tabellen oder in eienr HTML-Ansicht angezeigt und können als Excelliste heruntergeladen werden.

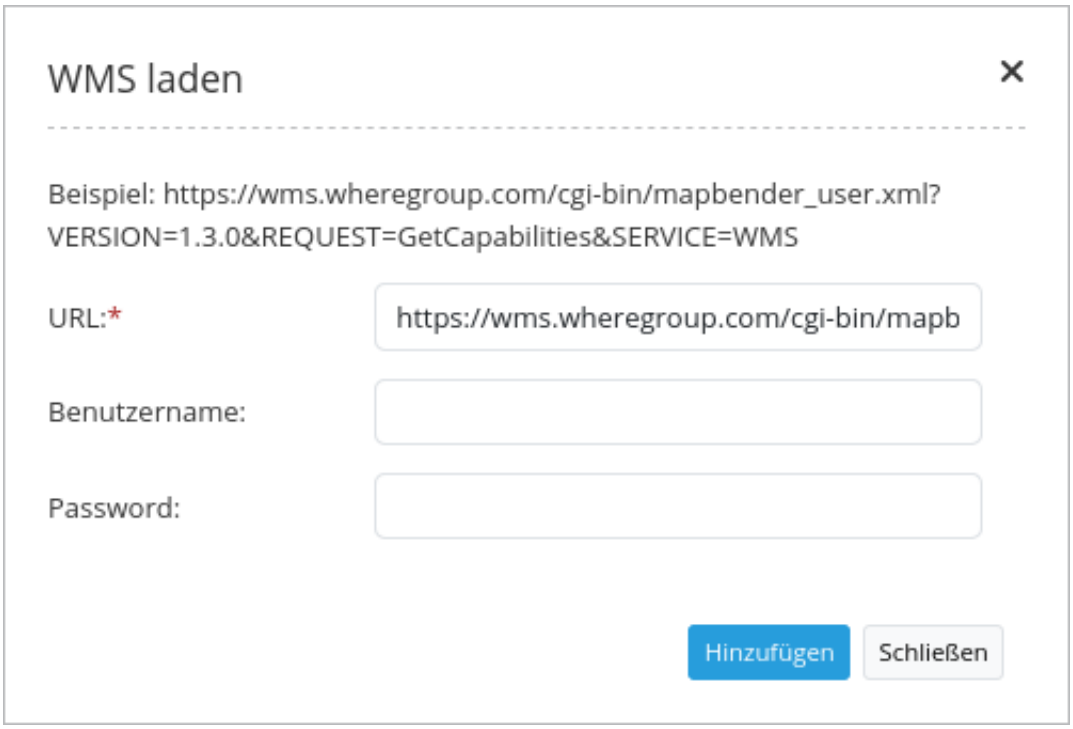
Weitere Informationen finden Sie unter: https://github.com/mapbender/query-builder/blob/master/README.md



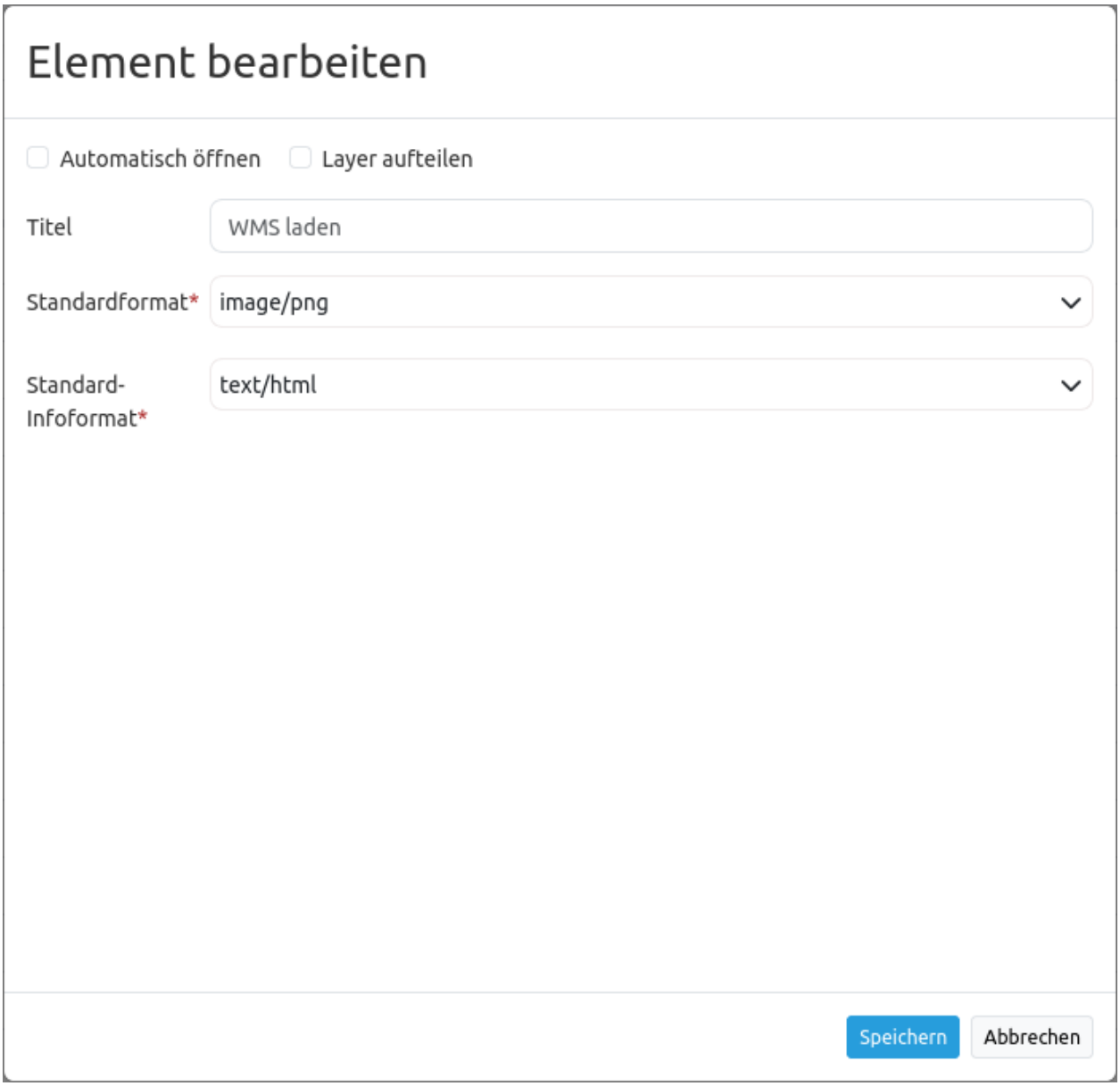
Konfiguration

3.6.12 WMS laden (WMS Loader)

Mit diesem Element können WMS per getCapabilities-Request in Mapbender geladen werden. Es kann WMS 1.1.1 und WMS 1.3.0 geladen werden.



Konfiguration



- **Automatisch Öffnen:** Öffnet Element beim Start der Anwendung (Standard: false).
- **Layer aufteilen:** Teilt Ebenen beim Laden des Dienstes auf (Standard: false).
- **Titel:** Titel des Elements. Dieser wird in der Layouts Liste angezeigt und ermöglicht, mehrere Button-Elemente voneinander zu unterscheiden. Der Titel wird außerdem neben dem Button angezeigt, wenn "Beschriftung anzeigen" aktiviert ist.
- **Standardformat:** image/png, image/gif, image/jpeg (Standard: image/png).
- **Standardinfoformat:** text/html, text/xml, text/plain (Standard: text/html).

Für das Element wird ein Button verwendet. Siehe unter *Button* für weitere Informationen zu dessen Konfiguration.

Hinzufügen eines WMS über einen definierten Link

Mapbender kann einen WMS über einen definierten Link hinzufügen, z. B. über *Information (FeatureInfo)* oder über Suchergebnisse. Der Link sollte folgendermaßen aussehen:

```
<a href="#"
mb-action="source.add.wms" mb-layer-merge="1" mb-wms-merge="1"
mb-wms-layers="Gewaesser,Fluesse"
mb-add-vendor-specific="bplan=123"
mb-infoformat="text/plain"
mb-url="http://wms.wheregroup.com/cgi-bin/germany.xml?VERSION=1.1.1&REQUEST=GetCapabilities&SERVICE=WMS
↪">load service</a>
```

```
mb-action="source.add.wms"         # Definiert die Aktion, um einen WMS hinzuzufügen
mb-wms-merge="1"                   # Fügt den WMS nur einmal hinzu, wenn der WMS in der Applikation
↪bereits Bestandteil ist, wird dieser verwendet (Standard: 1)
mb-layer-merge="1"                 # Aktiviert die Ebene in mb-wms-layers. Deaktiviert die Ebenen nicht,
↪ die schon aktiviert sind (Standard: 1).
mb-wms-layers="Gewaesser,Fluesse"  # Definiert die Ebenen, die aktiviert werden sollen, _all activates
↪aktiviert alle Ebenen.
href oder mb-url                   # Verweist auf die WMS getcapabilities URL.
mb-add-vendor-specific="bplan=123" # Definition von Vendor Specific Parametern, die an den Request
↪angefügt werden (ab Version 3.2.9).
mb-infoformat="text/plain"         # Definition des GetFeatureInfo-Formats für den WMS (Standard: text/
↪html)
```

YAML-Definition

Diese Vorlage kann genutzt werden, um das Element in einer YAML-Anwendung einzubinden.

```
target: ~                          # ID des Kartenelements
tooltip: 'WMS Loader'              # Text des Tooltips
autoOpen: false                    # true, wenn das Element beim Start der Anwendung geöffnet werden
↪soll (Standard: false).
defaultFormat: 'image/png'         # image/png, image/gif, image/jpeg (Standard: image/png)
defaultInfoFormat: 'text/html'     # text/html, text/xml, text/plain (Standard: text/html)
splitLayers: false                 # Nutzt geteilte Ebenen beim Laden des Dienstes (Standard: false)
```

CHAPTER

FOUR

BACKEND

Das Backend ist die Verwaltungsoberfläche von Mapbender: Mit ihm können Anwendungen und Datenquellen verwaltet und über Benutzer und Gruppen abgesichert werden. Zur Installation und Einrichtung von Mapbender besuchen Sie bitte die Seiten zur *Installation* bzw. den *Schnellstart*. In einer lokalen Standardkonfiguration kann das Backend über `http://localhost/mapbender` aufgerufen werden. Nach der erfolgreichen Anmeldung mit dem root-Benutzer stehen folgende Seiten zur Verfügung:

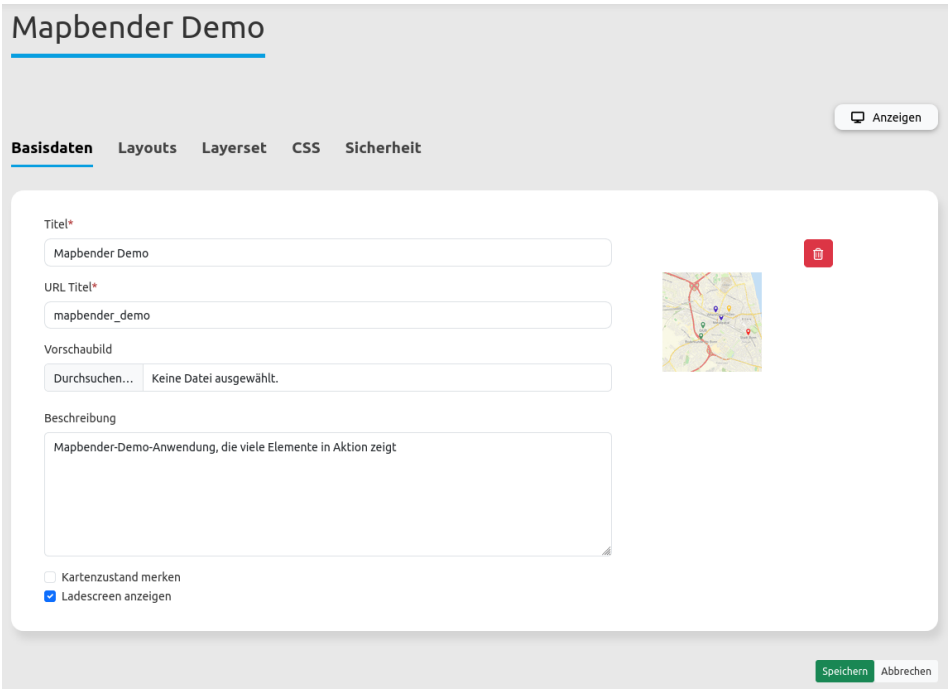
4.1 Anwendungen

4.1.1 Basisdaten

Die Basisdaten bestimmen die grundlegenden Informationen und Einstellungen einer Anwendung. Die Basisdaten sind über eine Maske in den Anwendungseinstellungen im *Backend* konfigurierbar. Mehr Details zu den verschiedenen Möglichkeiten der Anwendungserstellung finden Sie im *Schnellstart*.

Konfiguration

- **Titel**: Titel der Anwendung, frei ausfüllbares Textfeld.
- **URL Titel**: Titel der Anwendung als URL, frei ausfüllbares Textfeld. Keine Umlaute und Sonderzeichen erlaubt: URL Titel muss sich nach den Standards der festgelegten URL-Syntax richten.
- **Vorschaubild**: Aus dem Dateiverzeichnis hochladbare Bilddatei, die als Vorschaubild in der Anwendungsübersicht angezeigt wird. Klicken Sie auf den Button "*Datei auswählen*".
- **Beschreibung**: Beschreibung der Anwendung, frei ausfüllbares Textfeld.
- **Kartenzustand merken**: Speichert sitzungsübergreifend den Zustand bestimmter Kartenparameter und -einstellungen. Weitere Informationen finden Sie auf der Seite zu den *Share-Elementen*.
- **Ladescreen zeigen**: Zeigt ein Bild bei Anwendungsstart an, das die Ladezeit graphisch überbrückt.



Konfigurieren des Ladescreens

Der Ladescreen verbessert das Erscheinungsbild Ihrer Anwendung, indem er ein Logo und den Anwendungstitel anzeigt. Um den Ladescreen zu konfigurieren, befolgen Sie die folgenden Schritte:

1. Öffnen Sie die Datei `parameters.yaml` in Ihrer Mapbender-Installation. Weitere Informationen zur Datei selbst finden Sie unter *YAML Konfiguration (Konfigurations- und Anwendungsdateien)*.
2. Erstellen Sie den Schlüssel `branding.splashscreen_image` oder suchen Sie danach.
3. Definieren Sie den Ladescreen mithilfe einer der folgenden Methoden:
   - **Dateipfad**: Geben Sie einen einzelnen Dateipfad relativ zum Verzeichnis *application/public* in Ihrer Mapbender-Installation an.

```
parameters:
  branding.splashscreen_image: path/relative/to/public/myimage.png
```

   - **Array**: Verwenden Sie ein Array, bei dem die Schlüssel dem Slug der Anwendung entsprechen. Dies ermöglicht eine individuelle Anpassung des Ladescreens für verschiedene Anwendungen. Verwenden Sie den Schlüssel `default`, um ein Ersatzbild für Anwendungen bereitzustellen, die nicht explizit definiert sind.

```
parameters:
  branding.splashscreen_image:
    sample_application: path/relative/to/public/sample_application.png
    another_application: path/relative/to/public/another_application.png
    default: path/relative/to/public/myimage.png
```

Wenn keine individuelle Bilddatei für den Ladescreen konfiguriert wurde, wird stattdessen das Logo (`branding.logo`) für alle Anwendungen verwendet.

Darüber hinaus ist das Erscheinungsbild des Ladescreens über CSS-Variablen anpassbar. Bitte wechseln Sie zu *CSS-Editor* für ein Beispiel.

4.1.2 Layouts

Im Layouts-Bereich des *Backend* einer Anwendung werden die jeweiligen Frontend-Regionen der Anwendung und die in ihnen enthaltenen Elemente aufgelistet. Eine Übersicht über alle Elemente gibt es unter *Elemente*.

Note: Unterschiedliche Templates können durch unterschiedliche Regionen strukturiert sein.

Layout des Fullscreen Templates:

- Obere Werkzeugleiste (Region für die Platzierung von Buttons, Links, HTML, …)
- Sidepane (Seitenleisten-Region für den Ebenenbaum, die Legende, die Suche, den Druck, HTML, …)
- Kartenbereich (Region für die Karte, die Maßstabsleiste, …)
- Fußzeile (Region für das Impressum, die Aktivitätsanzeige, die Maßstabsauswahl, …)

Layout des Mobilen Templates:

- Fußzeile (Region für das Copyright, die Aktivitätsanzeige, die Maßstabsauswahl, …)
- Kartenbereich (Region für die Karte, die Maßstabsleiste, …)
- MobilePane (Region für Dialoge wie den Ebenenbaum, die Legende, den Hintergrundwechsler, die Infoabfrage, …)

Der + Button rechts oberhalb des Bereichs ermöglicht das Hinzufügen von Elementen. Nach dem Klick auf den Button öffnet sich eine Dialogmaske, die die Auswahl eines Elements und dessen anschließende Konfiguration ermöglicht.

Alle eingebauten Funktionen lassen sich per Drag & Drop zwischen den Regionen verschieben.

Die Regionen können über den Button in der oberen rechten Ecke individuell konfiguriert werden.

Die unterschiedlichen Regionen bieten folgende Konfigurationsmöglichkeiten:

Konfigurationsmöglichkeiten der Oberen Werkzeugleiste und Fußzeile

Die Regionen der Oberen Werkzeugleiste und der Fußzeile bieten folgende Konfigurationsmöglichkeiten über den Button an:

- **Bildschirmtyp** (Alle, Mobil, Desktop. Standard: Alle) Bei dieser Option wird der Bereich für die nicht ausgewählte Geräteart ausgeblendet. *Alle* zeigt die Region auf allen Geräten an.
- **Ausrichtung** (Links, Rechts, Zentriert. Standard: Rechts.): Die Ausrichtung definiert die Positionierung der Elemente innerhalb der Bereiche.
- **Checkbox Schaltflächen zu Menü zusammenfassen**: Konfiguriert ein Ausklappmenü, welches die in den Bereich eingebundenen Elemente umfasst.
- **Menütitel-Textfeld**: Mit dieser Textbox lässt sich dem Ausklappmenü eine Beschriftung zuweisen.

Tip: **Hinweis**: Das Ausklappmenü ist besonders sinnvoll, wenn die Anwendung für mobile Endgeräte ausgerichtet sein soll. Unter *CSS-Editor* findet sich ein Codebaustein, der die Bedienbarkeit bei Anwendungen mit vielen Elementen erhöht.

YAML-Konfiguration

Diese Vorlage kann zur **Eigenschaftenkonfiguration** der Oberen Werkzeugleiste in einer YAML-Anwendung verwendet werden:

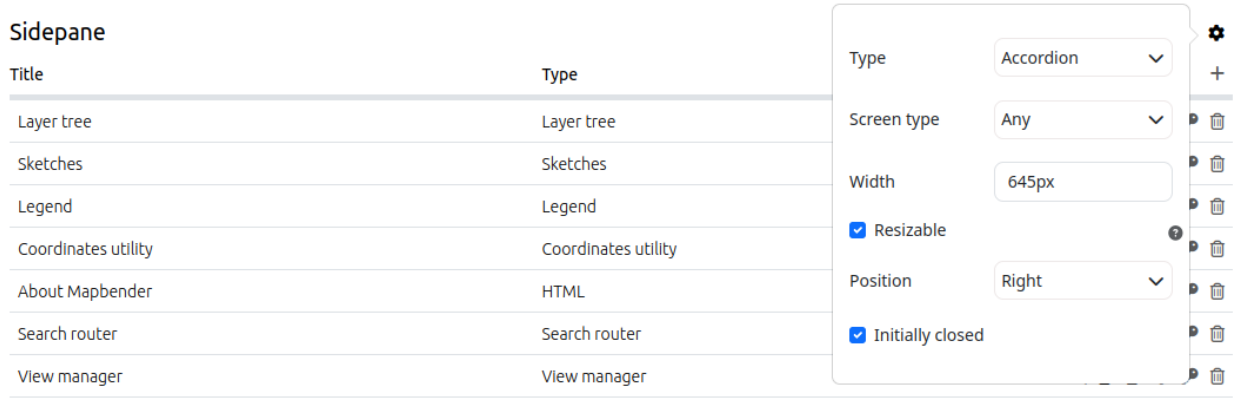
```
- name: toolbar
  properties:
    item_alignment: right  # right/left/center
    screenType: all         # desktop/mobile/all
    generate_button_menu: false # true/false
    menu_label: "Menu" # NULL or text
```

Außerdem kann diese Vorlage verwendet werden, um eine **Eigenschaftenkonfiguration** für die Fußzeile einer YAML-Anwendung vorzunehmen:

```
- name: footer
  properties:
    item_alignment: right # right/left/center
    screenType: all # all/desktop/mobile
    generate_button_menu: false # true/false
    menu_label: "Menu" # NULL or text
```

Sidepane-Konfigurationsmöglichkeiten

Die Ansichtsoptionen für die Sidepane können im Sidepane-Bereich im Mapbender-*Backend* ausgewählt werden. Dazu genügt ein Klick auf den Button:



- **Typ**: Siehe Funktionsbeschreibung unten.
- **Bildschirmtyp** (Alle, Mobil, Desktop. Standard: Alle) Bei dieser Option wird der Bereich für die nicht ausgewählte Geräteart ausgeblendet. *Alle* zeigt die Region auf allen Geräten an.

- **Breite** (in Pixeln, Standard: 350 px) Definiert über einen Pixelwert die Breite der Sidepane in der Anwendung.
- **Größe veränderbar** (Standard: true): Erlaubt das Verbreitern und Verkleinern der Seitenleiste.
- **Position** (Links, Rechts. Standard: Links) gibt an, ob die Sidepane am linken oder rechten Bildschirmrand angezeigt wird.
- **Geschlossen starten** (Standard: deaktiviert) hält nach Aktivierung die Sidepane bei Anwendungsstart eingeklappt. Sie ist über einen Button in der Anwendung nachträglich aus- & wieder einklappbar.

Die Option **Typ** zeigt die Sidepane-Elemente in unterschiedlichen Ansichten an:

- `Akkordeon` zeigt alle hinzugefügten Elemente in Reitern.
- `Buttons` zeigt alle hinzugefügten Elemente über Buttons.
- `Unformatiert` verzichtet auf Styling-Optionen und zeigt die Elemente direkt und in der im *Backend* gewählten Reihenfolge untereinander an.

Die Option **Größe verändern** geht von einer minimalen Breite von 120 px und einem Maximum von 95 % der Bildschirmbreite aus. Diese Werte können über CSS angepasst werden:

```
.sidePane.resizable {
  min-width: 200px;
  max-width: 500px;
}
```

YAML-Konfiguration

Diese Vorlage kann zur **Eigenschaftenkonfiguration** der Sidepane in einer YAML-Anwendung verwendet werden:

```
- name: sidepane
  properties:
    name: accordion # tabs (für Button) / accordion (für Akkordeon) / NULL (für unformatiert)
    align: right # right/left
    closed: false # true/false
    screenType: all # all/desktop/mobile
    width: "654px"
```

Element-Buttonleiste

Jedem Element kann ein konfigurierter Button zugewiesen werden. Die Buttonleiste dient der Konfiguration dieser Buttons. Dabei verfügen die Buttons über folgende Optionen:

- **aktiv/inaktiv**: Ein aktives Element ist in der Anwendung freigeschaltet. Ein inaktives Element kann zwar bearbeitet werden, ist aber in der Anwendung ausgeblendet.
- **Auf Mobilgeräten anzeigen**: Zeigt Elemente nur auf mobilen Geräten an.
- **Auf großen Bildschrimen anzeigen**: Zeigt Elemente nur auf Desktop-Geräten an.
- **Bearbeiten**: Erlaubt die Anpassung des Elements.

- **Zugriff auf das Element einschränken**: Erlaubt die Konfiguration von Element-Sichtbarkeiten bei bestimmten Berechtigungen.
- **Löschen**: Entfernt ein Element vom Back- und Frontend.

Bearbeiten

Öffnet über den Button die individuelle Konfigurationsmaske eines Elements. Diese finden Sie in der Dokumentation des jeweiligen Elements unter *Elemente*.

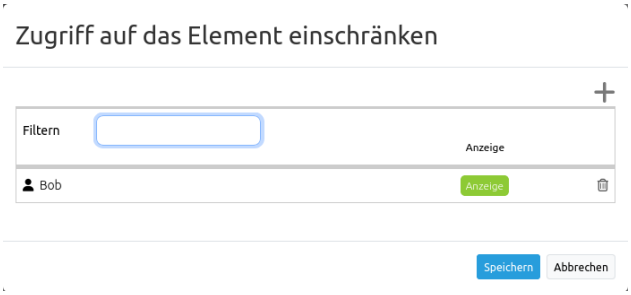
Zugriff auf das Element einschränken

Öffnet über den Button einen **Element sichern**-Dialog, der ein explizites Konfigurieren der *Ansehen*-Regel in *Rechteverwaltung* für Benutzer und Gruppen ermöglicht.

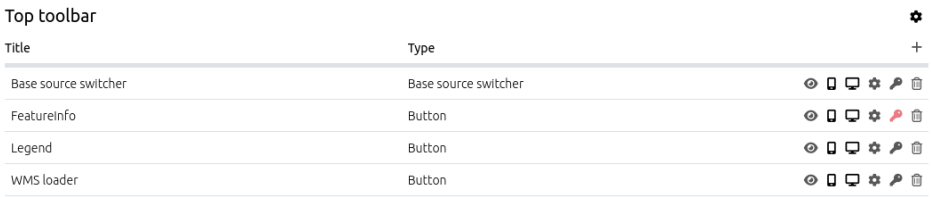
Ist dies nicht konfiguriert, hat das Element keine expliziten Zugangsbeschränkungen und sollte allen Benutzern und Gruppen zur Verfügung stehen.

Sofern die **View**-Regel für bestimmte Benutzer oder Gruppen gesetzt ist, können ausschließlich dort Eingetragene auf das Element zugreifen.

Beschränken Sie den Zugriff, indem Sie Benutzer über den Button hinzufügen. Eine gesetzte Checkbox zeigt an, dass die benötigten Berechtigungen zur Verfügung stehen.



Der Schlüssel wird nach erfolgreicher Rechtevergabe rot. Wenn Sie nun den Cursor über den Schlüssel halten, sehen Sie die Namen der berechtigten Nutzer in einem Pop-Up Fenster.



Detaillierte Informationen zu den Sicherheitseinstellungen finden sich unter *Rechteverwaltung*.

Löschen

Löscht das Element mitsamt der konfigurierten Einstellungen aus Front- und Backend.

4.1.3 Layerset

Ein Layerset ist ein logischer Container, der einen oder mehrere Layerset-Instanzen (WMS-Dienste) beinhalten kann. In den Demo-Anwendungen sind zwei Layersets definiert: Das Layerset "Main" für die Hauptkarte und das Layerset "Overview" für die Übersichtskarte. Die Namen der Layersets können frei gewählt werden. Außerdem können mehrere Layersets im Kartenelement ausgewählt werden. Zudem zeigt der Ebenenbaum den Namen der Layersets an, wenn die Option der thematischen Layer aktiviert ist.

Layerset-Übersichtsseite

Durch einen Klick auf **Layersets** wird die Übersichtsseite aufgerufen, auf der die Layersets konfiguriert werden. Jedes Layerset wird als Liste mit zugehörigen Diensten dargestellt. Die Übersichtsseite bietet folgende Funktionen:

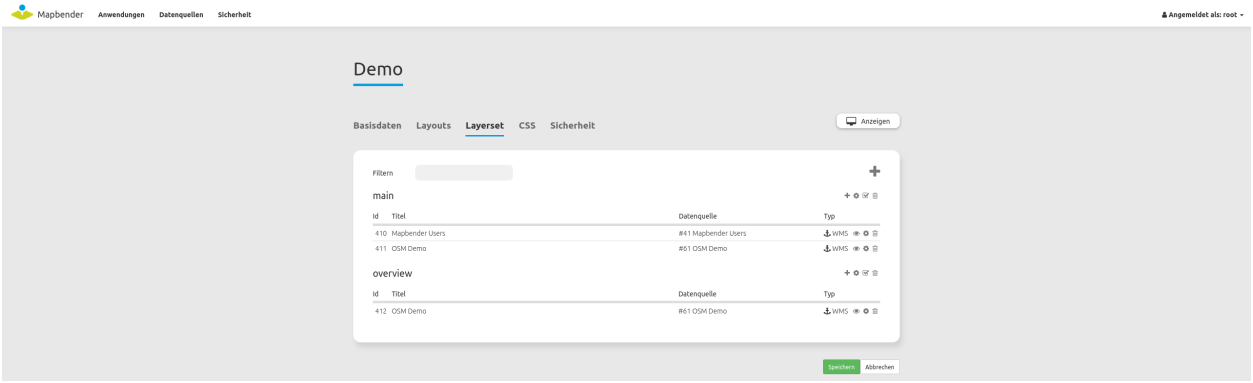
Layerset-Ebene

- : Layerset hinzufügen (großer Button)
- : Instanz hinzufügen (kleiner Button)
- : Layerset bearbeiten
- : Layerset beim Anwendungsstart aktivieren/deaktivieren
- : Layerset löschen

Instanz-Ebene

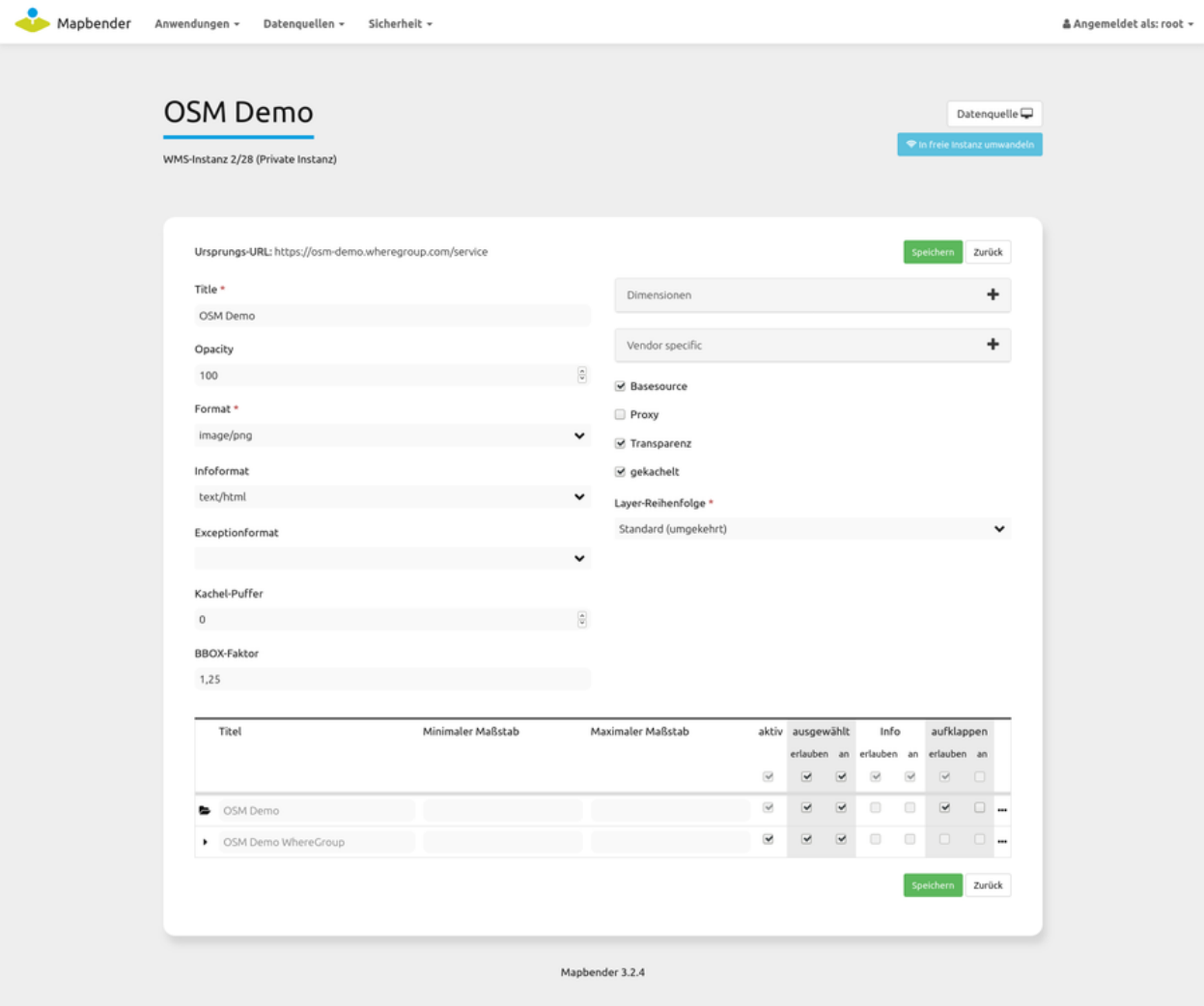
- : Instanz an/aus
- : Instanz bearbeiten

- : Instanz löschen



Layerset-Instanzen

Layerset-Instanzen enthalten unterschiedliche Optionen, mit denen WMS-Dienste angesprochen werden können: das Bildformat, das Infoformat, Exceptionformat, die Maßstäbe für die einzelnen Layer und vieles mehr. Aus der Layerset-Übersicht können mit dem Zahnrad-Button einzelne Instanzen konfiguriert werden.



Sobald ein WMS-Dienst in ein Layerset eingebunden wird, ist er als Layerset-Instanz mit der Anwendung verknüpft. Der Screenshot zeigt die *private Instanz* 2/28 basierend auf einem WMS-Dienst. Die zugehörige Datenquelle ist die Nr. 2. Die Layerset-Instanz selbst hat die Laufnummer 28.

Die Eigenschaften aus den WMS-Capabilities:

- **Title:** Der Titel der Instanz, der in der Liste der Layersets und ihrer Instanzen angezeigt wird.
- **Format:** Das Bildformat, mit dem die Kartenbilder über den GetMap-Request in der Anwendung abgerufen werden sollen. Für Rasterkarten und Luftbilder empfiehlt sich das JPG Format, für z.B. Straßenkarten das PNG Format. Im Zweifelsfall sollte PNG verwendet werden.
- **Infoformat:** Das Format, in denen die GetFeatureInfo Anfragen an den WMS abgeschickt werden soll. Im Zweifel: text/html oder ein analoges HTML-Format, was dann in den Dialog des *Information (FeatureInfo)* verwendet werden kann. Alternativ ist die Verwendung von text/plain möglich.
- **Exceptionformat:** Das Format für Fehlermeldungen aus den OGC Exception Formaten, welche der WMS anbietet.

Die Eigenschaften für die Anwendung

- **Opacity:** Die Deckkraft des Dienstes in Prozent. Dieser Wert ist nur dann für den Benutzer im *Ebenenbaum (Layertree)* konfigurierbar, wenn im Menü auch die “Opacity” aktiviert ist.

- **Kachel-Puffer (Tile buffer):** Dieser Parameter gilt für Dienste, die gekachelt angefordert werden und gibt an, ob weitere umgebende Kacheln abgerufen werden sollen. Damit sind diese bei einer Pan-Bewegung schon heruntergeladen und sichtbar. Je höher der Wert, desto mehr umgebende Kacheln werden abgerufen (Standard: 0).
- **BBOX-Faktor:** Dieser Parameter gilt für Dienste, die nicht gekachelt angefordert werden. Hier kann angegeben werden, wie groß das zurückgegebene Bild sein soll. Ein Wert größer 1 wird ein größeres Kartenbild anfordern (Standard: 1.25, kann auf 1 gesetzt werden).
- **Vendor Specific Parameters:** Siehe unten.
- **Sichtbarkeit:** Der Dienst kann mit dieser Schaltfläche sichtbar geschaltet werden.
- **BaseSource:** Der Dienst soll als BaseSource (Basisdienst) behandelt werden. Dies hat Auswirkungen auf den *Hintergrund wechseln*, der nur BaseSources anzeigen soll und auf den *Ebenenbaum (Layertree)*, in dem diese BaseSources dann auch ausgeblendet werden können. Siehe auch unter *Hinweise*.
- **Proxy:** Bei Aktivierung wird der Dienst über Mapbender als Proxy angefordert. Siehe auch unter *Hinweise*.
- **Transparenz:** Ist dieser Schalter aktiviert (und das ist er standardmäßig) wird der Dienst mit transparentem Hintergrund angefordert. Also im WMS GetMap-Request mit dem Parameter `TRANSPARENT=TRUE`.
- **Gekachelt (Tiled):** Der Dienst wird in Kacheln angefordert (Standard: nicht gekachelt). Siehe auch unter *Hinweise*.

Layer-Reihenfolge:

Es gibt zwei Möglichkeiten, wie die Layerreihenfolge dem Layerbaum übergeben wird:

- **Standard**
- **QGIS Style**

Diese Unterscheidung hat ihre Ursprünge in der Art und Weise wie WMS-Dienste und Capabilities-Dokumente aufgebaut sind.

Die OGC E-Learning Modul für Web Map Services beschreibt den Aufbau eines Beispiel-WMS.

Dieser WMS zeigt einige Punkt-, Linien- und Flächenobjekte und Rasterbilder an. Das Capabilities Dokument listet dabei die Bilddaten (z.B. DTM) weiter oben als die Punkte (z.B. osm_points als vorletzter Layer). Dieses Prinzip übernimmt Mapbender sowohl in den DataSources, wo der Dienst in Mapbender eingeladen und registriert wird, als auch in den Layerset-Instanzen.

Im Layerbaum dreht sich diese Reihenfolge dann aber um. Unserer Erfahrung nach vereinfacht dies die Nutzerfreundlichkeit. Denn somit werden Layer, welche weiter oben im Ebenenbaum positioniert sind, auch als oberste Layer angezeigt. Dieses Verhalten ist überlicherweise auch in einem Desktop-GIS zu finden. Punkte werden dann im Layerbaum weiter oben und Rasterdaten weiter unten dargestellt.

Der QGIS Server verhält sich hier aber anders. Ein WMS kann mithilfe der QGIS-Projektdatei sehr einfach auf einen QGIS Server konfiguriert werden. Die Reihenfolge der Layer in den WMS Capabilities ist dann gleich der Reihenfolge, die im eigenen QGIS-Projekt genutzt wird. Es gilt generell: Punkte oben, darunter die Linien, dann zum Schluß die Polygondaten oder Rasterbilder. QGIS und QGIS Server stellen dabei nicht die einzigen Programme dar, welche dies so machen. Sie sind aber die besten. Für den GetMap Aufruf sind diese Einstellungen dann wieder irrelevant. Denn die WMS-Spezifikation klärt bereits die Art und Weise der Aufrufe und wie diese an den Klienten zurückgesendet werden.

Die folgende Tabelle fasst das Verhalten nochmal zusammen:

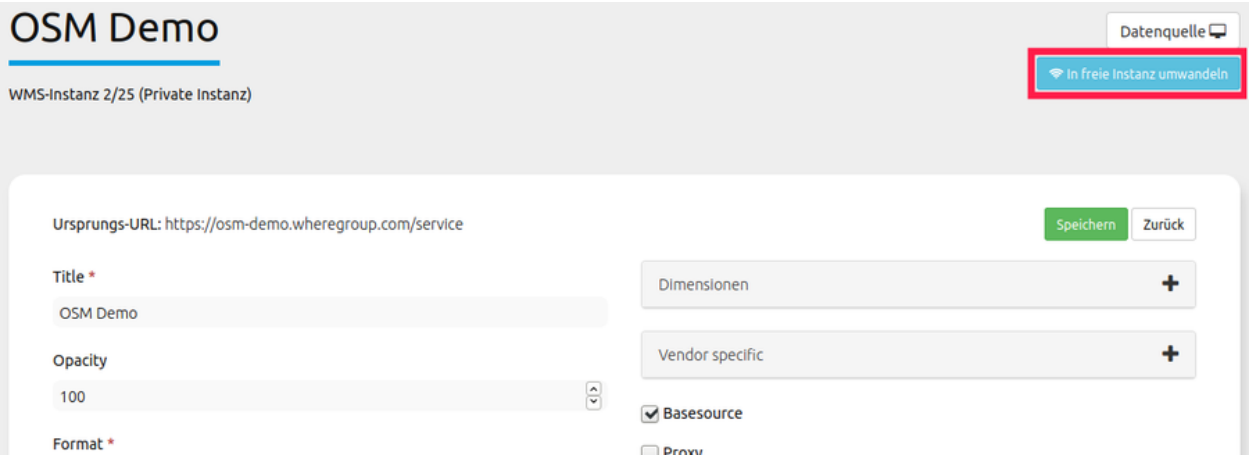
	layer order standard	layer order QGIS style
WMS Capabilities (from top to bottom)	polygon, line, point	point, line, polygon
Layerset-Instance (from top to bottom)	polygon, line, point	point, line, polygon
Layertree (from top to bottom)	point, line, polygon	point, line, polygon

Damit kann Mapbender auf die unterschiedlichen Art und Weisen reagieren, die ein WMS Capabilities Dokument aufgebaut hat, indem einfach die Reihenfolge in dem Layerbaum angepasst wird.

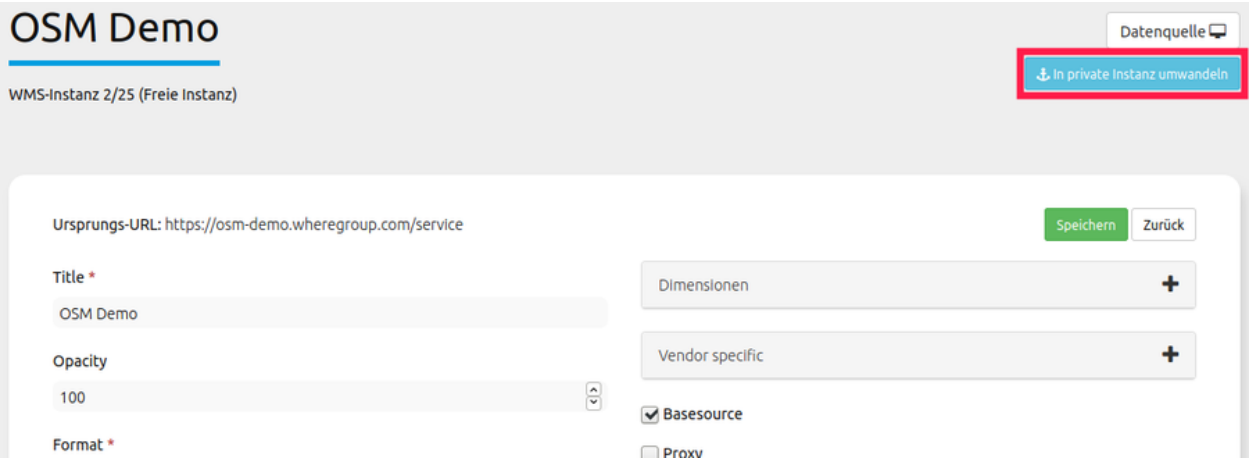
Freie und private Instanzen

Alle Layerset-Instanzen werden standardmäßig als private Instanzen erstellt. Private Instanzen müssen für jede Anwendung individuell konfiguriert werden. Freie Instanzen ermöglichen die Einbindung einer vorkonfigurierten Layerset-Instanz in mehreren Anwendungen. Freie Instanzen können verwendet werden, um für mehrere Anwendungen, welche gleich konfigurierte Layerset-Instanzen teilen, nicht mehrmals die gleiche Konfiguration durchführen zu müssen. Änderungen einer freien Instanz sind in allen Anwendungen, in denen die Instanz eingebunden ist, wirksam.

Eine Layerset-Instanz kann in der Bearbeitungsansicht über den blauen Button “In freie Instanz umwandeln” umgeändert werden. Über den gleichen Button lässt sich eine freie Instanz wieder in eine private Instanz umwandeln.

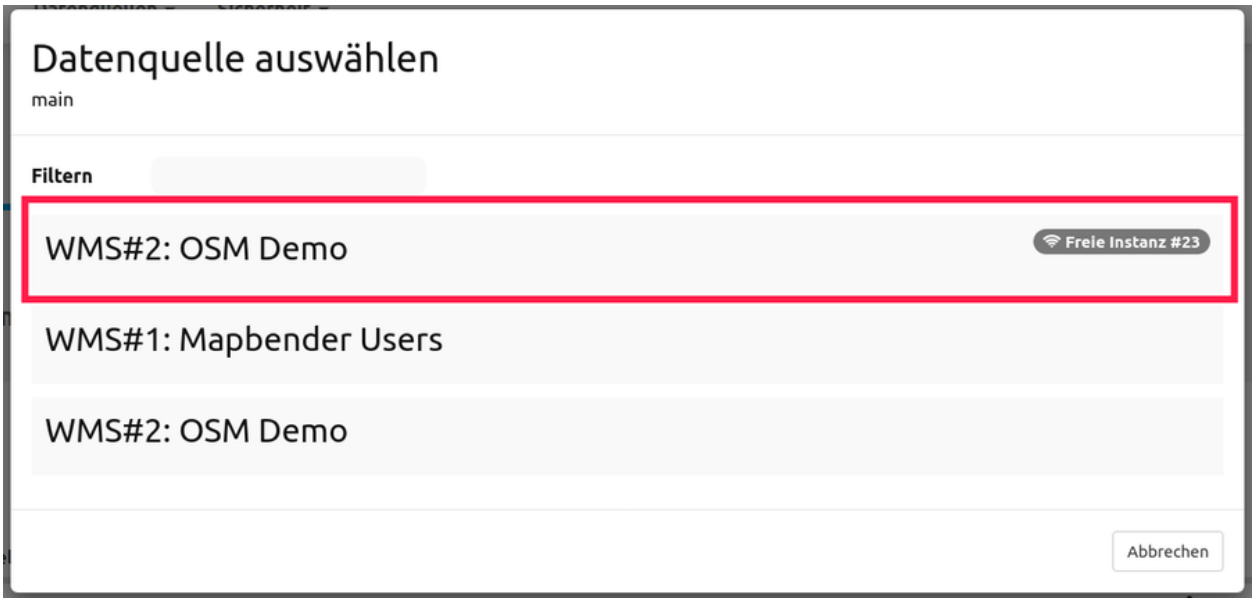


In der Layersets-Übersicht sind freie Instanzen in der Spalte “Typ” mit einem Funk-Symbol gekennzeichnet.



Private Instanzen sind wiederum mit einem Anker-Symbol gekennzeichnet.

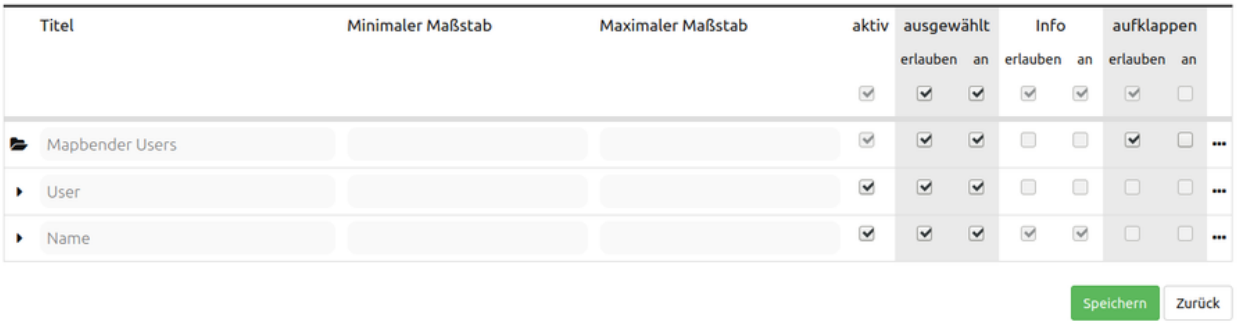
Um eine freie Instanz in eine Anwendung einzubinden, kann diese als Datenquelle beim Hinzufügen einer Instanz ausgewählt werden.



Zu einer Übersicht aller freien Instanzen kann über das Menü zu **Datenquellen -> Freie Instanzen** navigiert werden.

Layerkonfiguration

Über die verschiedenen Checkboxen können Sie verschiedene Einstellungen treffen. Beachten Sie dabei, dass die oberste Checkbox der Liste jeweils für alle Instanzen denselben Status auswählt.



- **Titel**: Layertitel der Service Information (anpassbar).
- **Minimaler/Maximaler Maßstab**: Der Maßstabsbereich, in dem der Layer angezeigt wird.
- **Aktiv an/aus**: Aktiviert/deaktiviert ein Thema in dieser Anwendung. Sobald dieser Haken nicht gesetzt ist, werden alle anderen Haken derselben Instanz ignoriert.
- **Auswählen erlauben**: Der Layer ist im Ebenenbaum auswählbar.
- **Auswählen an**: Der Layer ist bei Anwendungsstart im Ebenenbaum aktiv.

- **Info erlauben**: Die Infoabfrage wird für diesen Layer zugelassen.
- **Info an**: Die Infoabfrage wird beim Start aktiviert.
- **Aufklappen erlauben**: Der Layer ist über den Ebenenbaum aufklappbar.
- **Aufklappen an**: Aufklappen des Layers beim Start der Anwendung.
- **Sortieren**: Die Ebenen können über Drag & Drop in der Anwendung verschoben werden.
- **...** (Drei-Punkte-Menü): Öffnet einen Dialog mit weiteren Informationen:
- **Name**: Layername der Service Information (wird beim getMap-Request verwendet und ist nicht veränderbar).
- **Style**: Wenn ein WMS mehr als einen Stil anbietet, können Sie einen anderen Stil als den Standard-Stil wählen.

Hinweise zu den Auswirkungen der einzelnen Konfigurationen

Basesources

Es gibt verschiedene Möglichkeiten, den Ebenenbaum zu füllen und mit Basesources zu arbeiten:

- Durch das Verstecken im Layerbaum und das Nutzen des *Hintergrund wechseln*-Elements.
- Alternativ kann der *Workflow thematischer Ebenenbaum* genutzt werden. Dieser zeigt den Namen des Layersets entlang eines Ordners und einer Checkbox im Ebenenbaum. Dabei lässt sich konfigurieren, ob der Ordner nach dem Öffnen der Anwendung angezeigt werden soll; zudem ist es möglich, den Ordner aufgeklappt darzustellen.

Proxy

Wozu dient dieser Schalter? Die Proxynutzung macht Sinn, wenn vermieden werden soll, dass der Webbrowser als Client direkt auf den Dienst zugreift (dies ist die OpenLayers Standardeinstellung). Ist der Schalter jedoch aktiviert, greift Mapbender mit seiner URL auf den Dienst zu, verarbeitet die Bilder und stellt sie in der Karte dar. D.h. es kann über Firewalls sehr leicht ein über das Netzwerk geschützter Dienst angeboten werden, der nur von dem Webserver angesprochen werden darf, auf dem Mapbender läuft.

Kachelung, Kartengröße und Performance

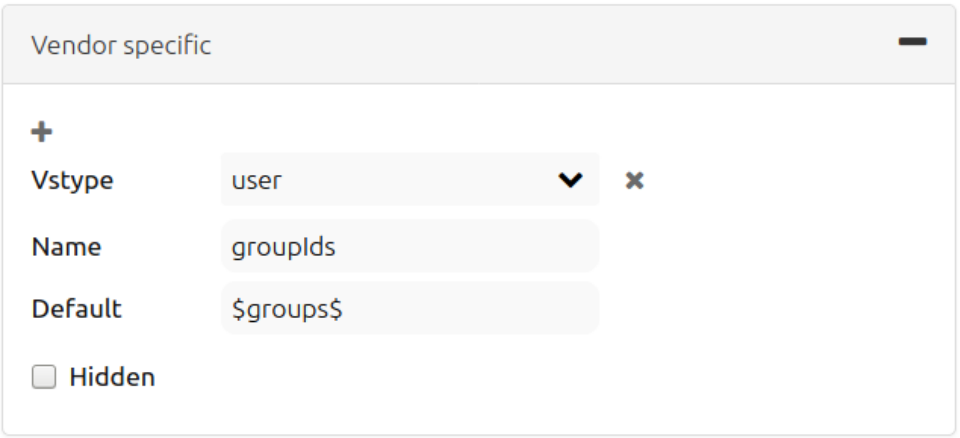
Der Parameter "Gekachelt" wird benutzt, um das Kartenbild in einzelnen Kacheln anzufordern und nicht als ganzes Bild. Bei Nutzung eines Mapproxy , der den Dienst gekachelt bereitstellt, sollte der Parameter eingeschaltet werden. Es macht aber auch für normale, ungekachelte Dienste Sinn, da die gefühlte Wartezeit beim Nutzer geringer ist: Das Kartenbild erscheint, obwohl noch nicht alle Kacheln abgerufen worden sind.

Man muss dabei aber beachten: Die Anzahl der Anfragen an einen WMS vergrößert sich rapide: Je nach Bildschirm-Auflösung und eingestellter Kachelgröße in der *Karte (Map)* werden statt einer viele Anfragen abgeschickt. Die zurückgelieferten Bilder sind zwar nicht besonders groß, aber zahlreich. Auch in Hinblick auf die Verwendung des **Kachel-Puffers**. Es ist somit eine Abwägungs- und Fall-zu-Fall-Unterscheidung, wie der Dienst angesprochen werden soll. Die Performance kann auch über eigene Maßstabsangaben der Layer gesteigert werden, wenn der Dienst diese etwas zu locker vorgegeben hat.

Es gibt des Weiteren noch hier und da WMS-Dienste, die nur eine maximale Kartenbildgröße unterstützen und mit den hohen Auflösungen, die Mapbender anfragen kann, nicht zurechtkommen. Das Fullscreen-Template kann auf die maximale Bildschirmbreite gezogen werden und das angeforderte Kartenbild ist dann in etwa der Breite und Höhe des sichtbaren Browserfensters.

Vendor Specific Parameter

In einer Dienstinstanz können Vendor Specific Parameter angegeben werden, die an den WMS Request angefügt werden. In Mapbender können die Vendor Specific Parameter genutzt werden, um Benutzer und Gruppeninformation des angemeldeten Benutzers an die WMS-Anfrage zu hängen. Es können auch feste Werte übermittelt werden. Das folgende Beispiel zeigt die Definition eines Parameters „group“, der als Inhalt die Gruppe des gerade in Mapbender angemeldeten Nutzers weitergibt.



- Name: Parameter Name im WMS Request.
- default: Standardwert
- Vstype: Mapbender-spezifische Variablen: Gruppe (groups), User (users), Simple.
- Hidden: Wenn dieser Wert gesetzt wird, werden die Anfragen serverseitig versendet, sodass die Parameter nicht direkt sichtbar sind.

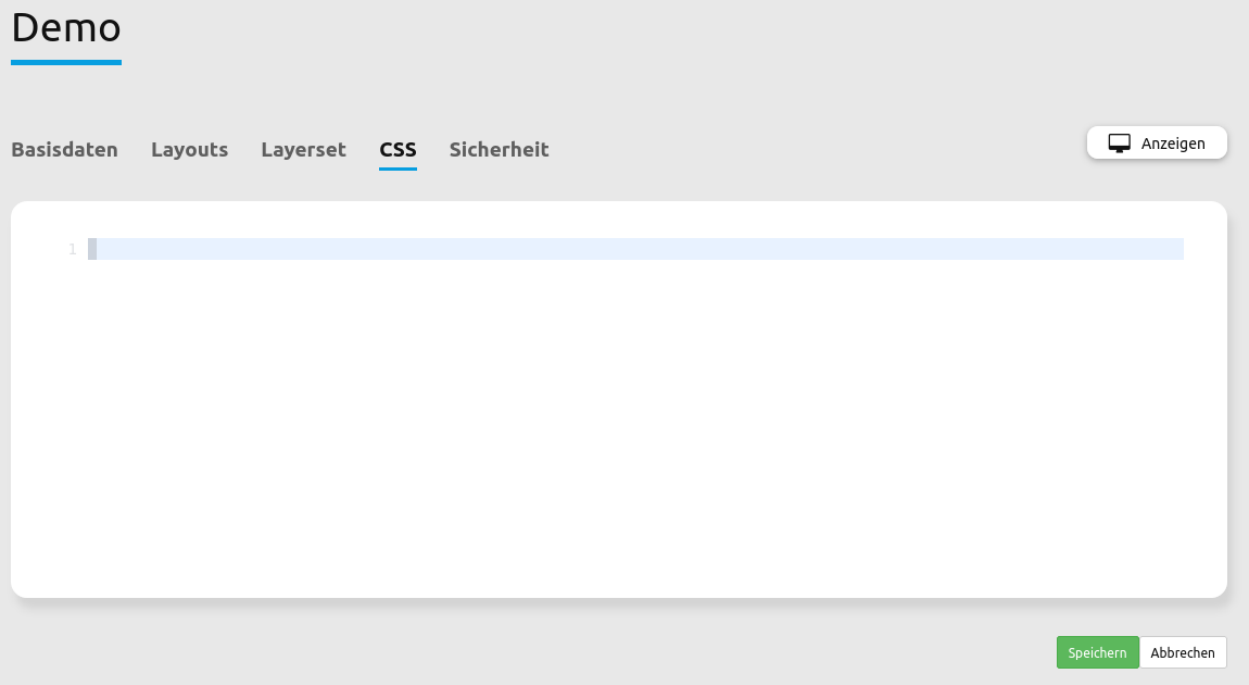
Momentan eignet sich das Element, um den Dienst nur an bestimmte Benutzer und Gruppen weiterzugeben. Dies geschieht z.B. für Benutzer über die `$id$` und für Gruppen über den Parameter `$groups$`.

Weitere Informationen

- Information zur Benutzung von Layersets finden Sie auch im Schnellstart-Kapitel *4. Datenquellen (Sources) verwenden*.
- Tipps, wie sich das Kacheln von WMS begrenzen lässt, finden Sie unter *Performance*.
- Die Bedeutung der Layersets für die Anzeige im Ebenenbaum ist im Abschnitt zu den Thematischen Layern wird im *Workflow thematischer Ebenenbaum* beschrieben.
- Einzelne Layersets können in den Einstellungen zur *Karte (Map)* für die jeweilige Anwendung an- bzw. abgeschaltet werden.
- Ebenso können bei der Anzeige thematischer Layer die Layersets in der Layerset-Konfiguration per Checkbox bei Anwendungsstart (de-)aktiviert werden.

4.1.4 CSS-Editor

Mapbender verfügt über einen CSS-Editor zur Anpassung des Stils (Farben, Größen, Icons, …) einer Anwendung. Es ist möglich, zusätzliche CSS-Klassen zu definieren, die den Standardstil überschreiben. Sie können außerdem SCSS im Editor verwenden und auf eine SCSS-Datei in Ihrem Bundle verweisen.



Tip: Mithilfe von Browser-Entwicklerwerkzeugen ist es möglich, Elemente zu identifizieren, ihre Klassen in den CSS-Editor zu kopieren und dort anzupassen.

Anwendungsbeispiel

Bildlaufleiste zu Menü hinzufügen

Wenn Sie das Schaltflächenmenü in der oberen Werkzeugleiste verwenden, können Sie festlegen, bei welcher Bildschirmgröße die Höhe reduziert und eine Bildlaufleiste angezeigt werden soll. Dies verbessert die Benutzerfreundlichkeit der Anwendung auf mobilen Geräten.

```
// Scrollbar in Toolbox
.dropdown-menu {
  overflow-y: auto;
  max-height: calc(100vh - 100px);
}
```

Die Menü-Funktion kann im **Layouts**-Tab des Backends gefunden werden: Klicken Sie auf den Zahnrad-Button und aktivieren anschließend das Kontrollkästchen **Menü für Schaltflächen generieren**.

Den Ladescreen individuell anpassen

Sofern der Ladescreen einer Anwendung über *Basisdaten* aktiviert ist, ist es möglich, diesen weiter über CSS anzupassen.

```
:root {
    --primary: #079ee0;                                 /* Farbe des Ladeindikators */
    --splashscreen-border: none;                        /* Rand um den Startbildschirm-Dialog, der z.B.
↪mit https://html-css-js.com/css/generator/border-outline/ generiert werden kann */
    --splashscreen-border-radius: 25px;                 /* Radius der abgerundeten Ecken um den Dialog
↪*/
    --splashscreen-background: rgba(255,255,255,0.8);   /* Hintergrundfarbe des Ladescreens */
    --splashscreen-fade-out-duration: 200ms;            /* Animationsdauer beim Ausblenden des
↪Ladescreens */
}
```

4.1.5 Sicherheit (Anwendungsreiter)

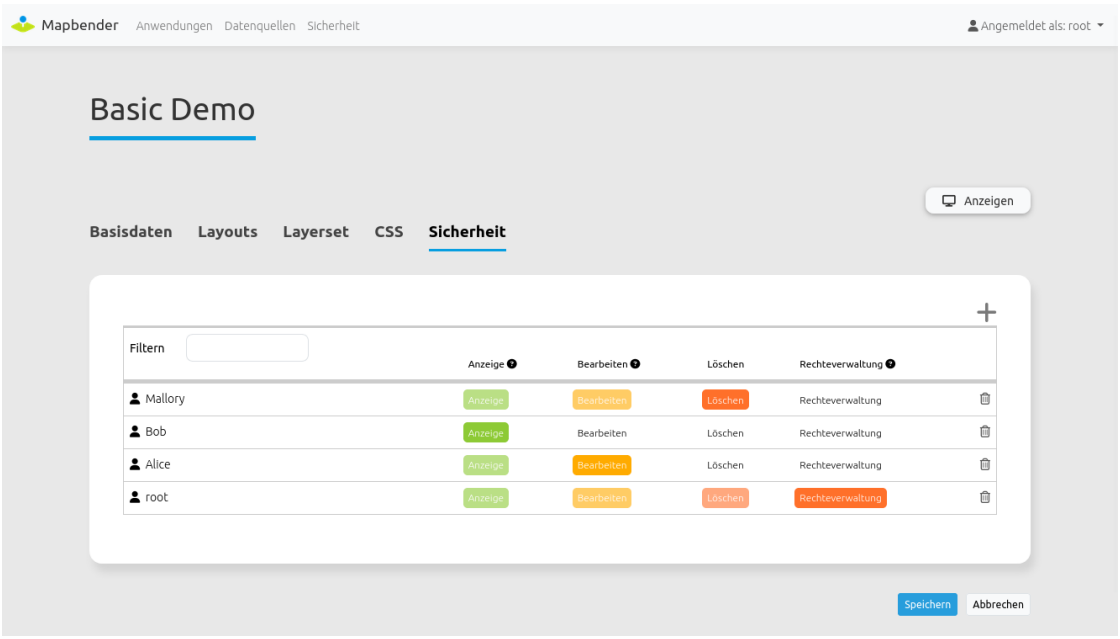
Jede Anwendung besitzt einen Reiter für individuelle Sicherheitseinstellungen. In diesen können Sie unterschiedliche Berechtigungen explizit für die individuelle Anwendung zuweisen.

Note: Eine globale gesetzte Berechtigung überschreibt eine über diese Seite gesetzte individuelle Berechtigung.

Anwendungszuweisungen

Standardmäßig stehen alle Anwendungen den *Benutzern* und *Gruppen* zur Verfügung, die mit ausreichenden Berechtigungen Zugriff auf das *Backend* haben. Für einzelne Anwendungen kann der Zugriff noch genauer definiert werden, so dass diese nur bestimmten Accounts zur Verfügung stehen.

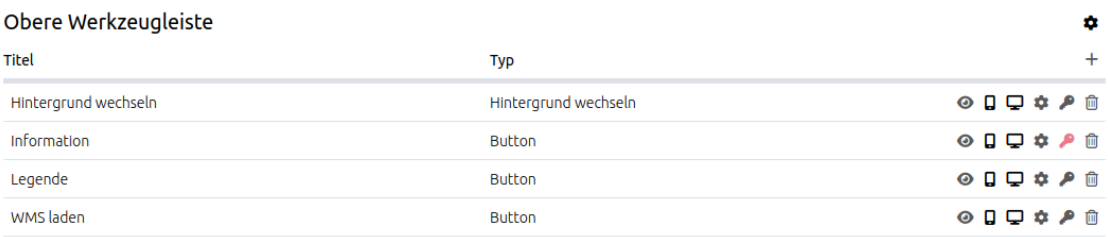
1. Bearbeiten Sie Ihre **Anwendung** .
2. Wählen Sie den Reiter **Sicherheit**.
3. Machen Sie ihre Anwendung öffentlich zugänglich über die Auswahl **Öffentlicher Zugriff** (oder alternativ über den **Anwendung publizieren/verbergen** in der Anwendungsübersicht).
4. Fügen Sie für individuelle Einstellungen alternativ Benutzer oder Gruppen über den Button hinzu. Setzen Sie anschließend individuelle Berechtigungen über die Berechtigungstabelle. So weisen Sie eine Anwendung einem oder mehreren Benutzern oder Gruppen zu.
5. Melden Sie sich erneut unter der ausgewählten Benutzerbezeichnung an, um die Rechtevergabe zu testen.



Elementzuweisungen

Standardmäßig stehen alle Elemente einer Anwendung den *Benutzern* und *Gruppen* zur Verfügung, die Zugriff auf sie haben. Für einzelne Elemente kann der Zugriff noch genauer definiert werden, so dass diese nur bestimmten Accounts zur Verfügung stehen.

1. Bearbeiten Sie Ihre **Anwendung** .
2. Wählen Sie den Reiter **Layouts**.
3. Jedes Element verfügt über einen eigenen Button. Wählen Sie den Button zu dem Element, das nur ausgewählten Benutzern oder Gruppen zur Verfügung stehen soll.
4. Fügen Sie Benutzer oder Gruppen über den Button hinzu. Setzen Sie anschließend die *view*-Berechtigung. Das Element wird so innerhalb der Anwendung abgesichert und nur den gewählten Benutzern oder Gruppen zugänglich.
5. Testen Sie die Konfiguration, indem Sie die Anwendung mit Benutzern aufrufen, die (keine) Berechtigungen zum Element erhalten haben.



4.2 Datenquellen (Sources)

Über den Backend-Menübereich Datenquellen können OGC WMS- und WMTS/TMS-Dienste in den Versionen 1.1.1 und 1.3.0 registriert werden.

Informationen zum Einbinden von Diensten und die Nutzung in Anwendungen finden Sie im Schnellstart-Kapitel *Laden von Datenquellen*.

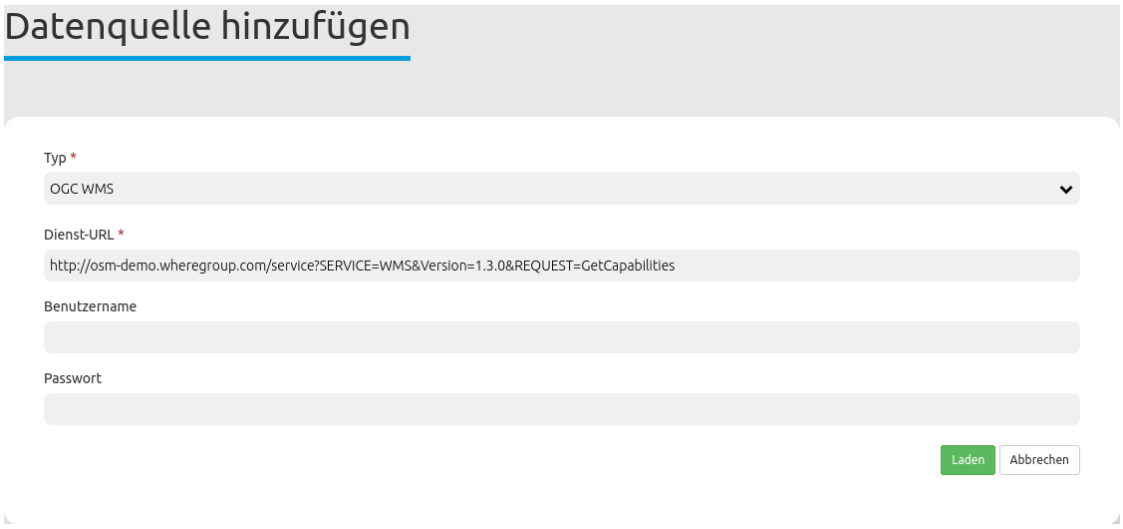
4.2.1 Datenquelle laden

Tip: Hinweis: Es ist wichtig, dass die Datenquelle vor dem Hochladen auf ihre Richtigkeit überprüft wird. Dies erfolgt über den Aufruf des getCapabilities-Requests im Browser.

Um einen Dienst zu laden, drücken Sie auf + **Datenquelle hinzufügen**. Dies öffnet einen Konfigurationsbereich mit folgenden Parametern:

- **Typ**: Dropdown-Auswahl zwischen Datentyp OGC WMS und OGC WMTS / TMS (Pflichtangabe).
- **Dienst-URL**: URL zum Capabilities-Dokument des Dienstes (z. B. für OGC WMS Version 1.3.0)
- **Benutzername / Passwort**: Eingabe von Benutzername und Passwort bei gesicherten Diensten.

Mit einem Klick auf Laden wird der Dienst registriert.

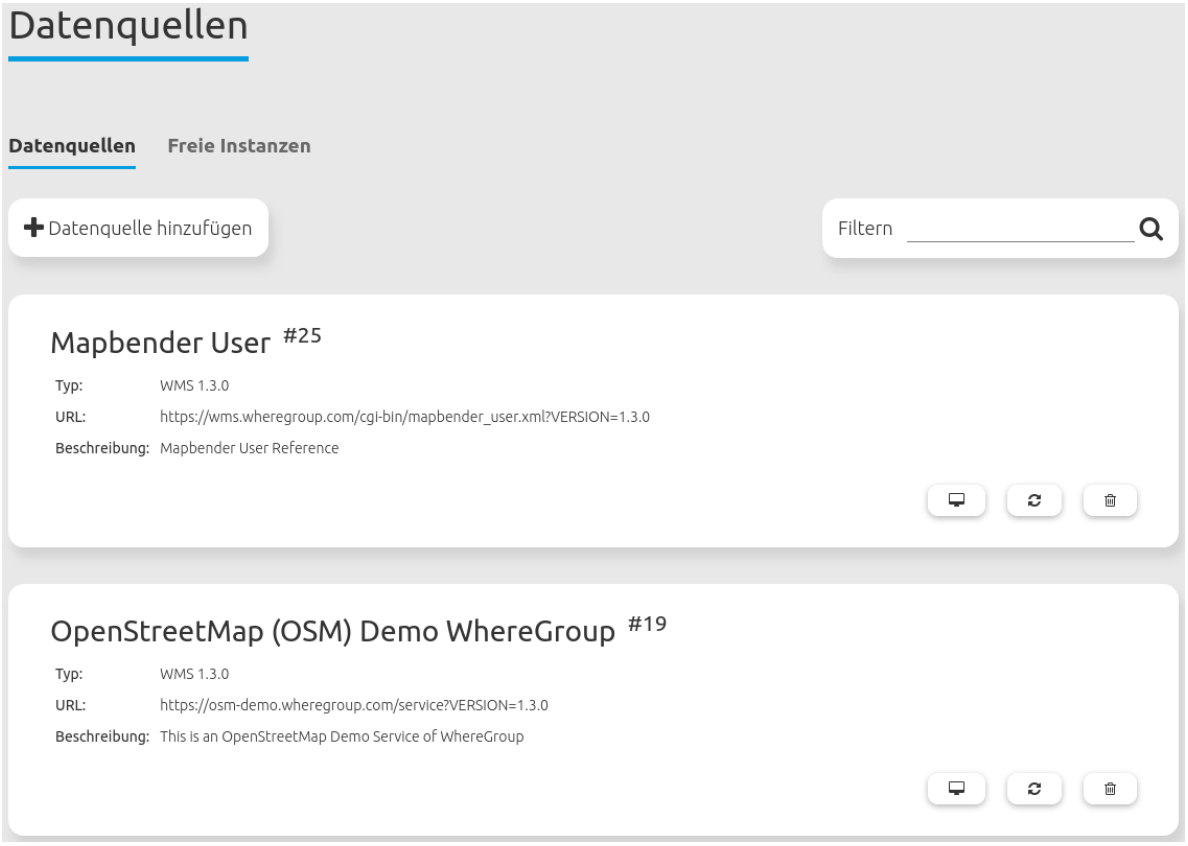


Nach einer erfolgreichen Dienstregistrierung zeigt Mapbender Informationen zum Dienst in einem Übersichtsfenster an.

4.2.2 Datenquellen-Übersicht

Die Bereiche Datenquellen und Freie Instanzen listen die erfolgreich geladenen und konfigurierten Dienste auf:

- **Filter**: Filtert die Dienste nach kontextspezifischer Eingabe, berücksichtigt Name, URL, Typ und Beschreibung.
- **Metadaten anzeigen**: Zeigt die Metadaten eines ausgewählten Dienstes an. Öffnet einen neuen Bereich, der in mehreren Reitern Metadaten, Mapbender-Anwendungen mit Zugriff, Kontaktinformationen, Details (z.B. Version) und die Layer des Dienstes ausgibt.
- **Datenquelle aktualisieren**: Aktualisiert die Dienst-Informationen durch erneutes Laden des getCapabilities-Dokuments.
- **Datenquelle entfernen**: Entfernt den Dienst aus Mapbender.



4.2.3 Datenquellen-Kontextmenü

Im Metadatendialog eines Dienstes befindet sich oben rechts außerdem das Datenquellen-Kontextmenü. Es ermöglicht folgende Funktionen:

- **Datenquelle aktualisieren**: Aktualisiert die Dienst-Informationen durch erneutes Laden des getCapabilities-Dokuments.
- **Freie Instanz erzeugen**: Erzeugt eine freie Instanz aus der Datenquelle. Diese wird im Bereich "Freie Instanzen" angezeigt.
- **Löschen**: Entfernt die freie Instanz aus Mapbender.

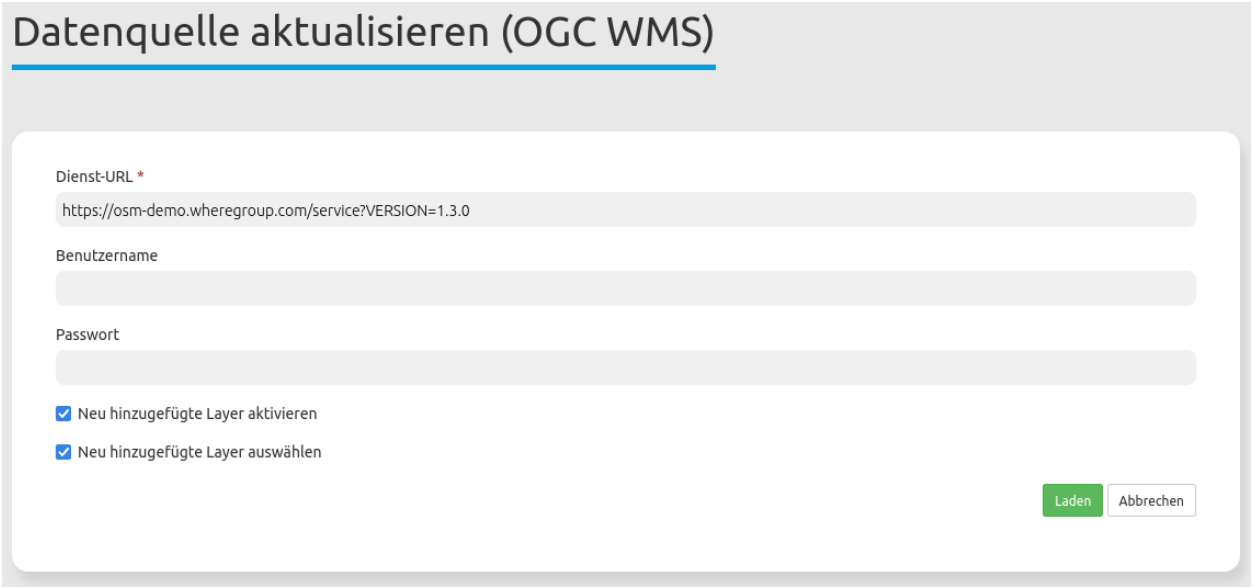
4.2.4 Datenquellen aktualisieren

Die Aktualisierung einer Datenquelle erfolgt zunächst über den Aufruf der Seite `Datenquellen` im Backend. Wählen Sie aus der Liste die zu aktualisierende Datenquelle aus. Es ist möglich, die Liste anhand des Suchfelds nach Diensten zu filtern. Klicken Sie anschließend neben der gewünschten Datenquelle auf den

Datenquelle aktualisieren-Button. Dadurch öffnet sich die Aktualisierungsmaske. Hier können Sie auch die URL oder Benutzername / Passwort des Dienstes anpassen.

Hint: Datenquellen lassen sich auch aktualisieren, ohne dass Änderungen vorgenommen wurden. Das Capabilities-Dokument wird neu eingelesen.

Zusätzlich bietet die Maske zwei Checkboxen an:



- **Neu hinzugefügte Layer aktivieren**: Ist der Haken an dieser Checkbox gesetzt, sind durch die Aktualisierung neu geladene Dienst-Layer automatisch in Anwendungen aktiv. Ist der Haken nicht

gesetzt, erscheinen neue Layer nicht im Ebenenbaum.

- **Neu hinzugefügte Layer auswählen**: Ist der Haken an dieser Checkbox gesetzt, werden durch die Aktualisierung neu geladene Dienst-Layer automatisch in Anwendungen sichtbar und sind aktiv. Dazu muss allerdings auch `Neu hinzugefügte Layer aktivieren` gesetzt sein. Ist `Neu hinzugefügte Layer auswählen` nicht gesetzt, erscheint der Layer zwar im Ebenenbaum, ist aber nicht aktiviert.

Falls die Änderungen vorgenommen werden sollen, klicken Sie auf den `Laden`-Button, um die Datenquelle zu aktualisieren. Dabei wird das getCapabilities-Dokument neu ausgelesen. Die aktualisierte Version wird anschließend in den Konfigurationseinstellungen angezeigt und Änderungen werden in Anwendungen, in denen der Dienst verwendet wird, angewandt.

4.3 Sicherheit

Die Absicherung von Domain-Objekten ist in Mapbender über ein Berechtigungssystem implementiert. Es erlaubt globale Berechtigungen auf Bereiche und individuelle Berechtigungen auf Objekte. Eine flexible Rechtezuweisung ist in diesen Bereichen möglich:

4.3.1 Benutzer

Benutzer werden als FOM\UserBundle\Entity\User implementiert und im Datenbank Repository gespeichert. Die Entität hält nur die notwendigen Informationen über einen Nutzer vor.

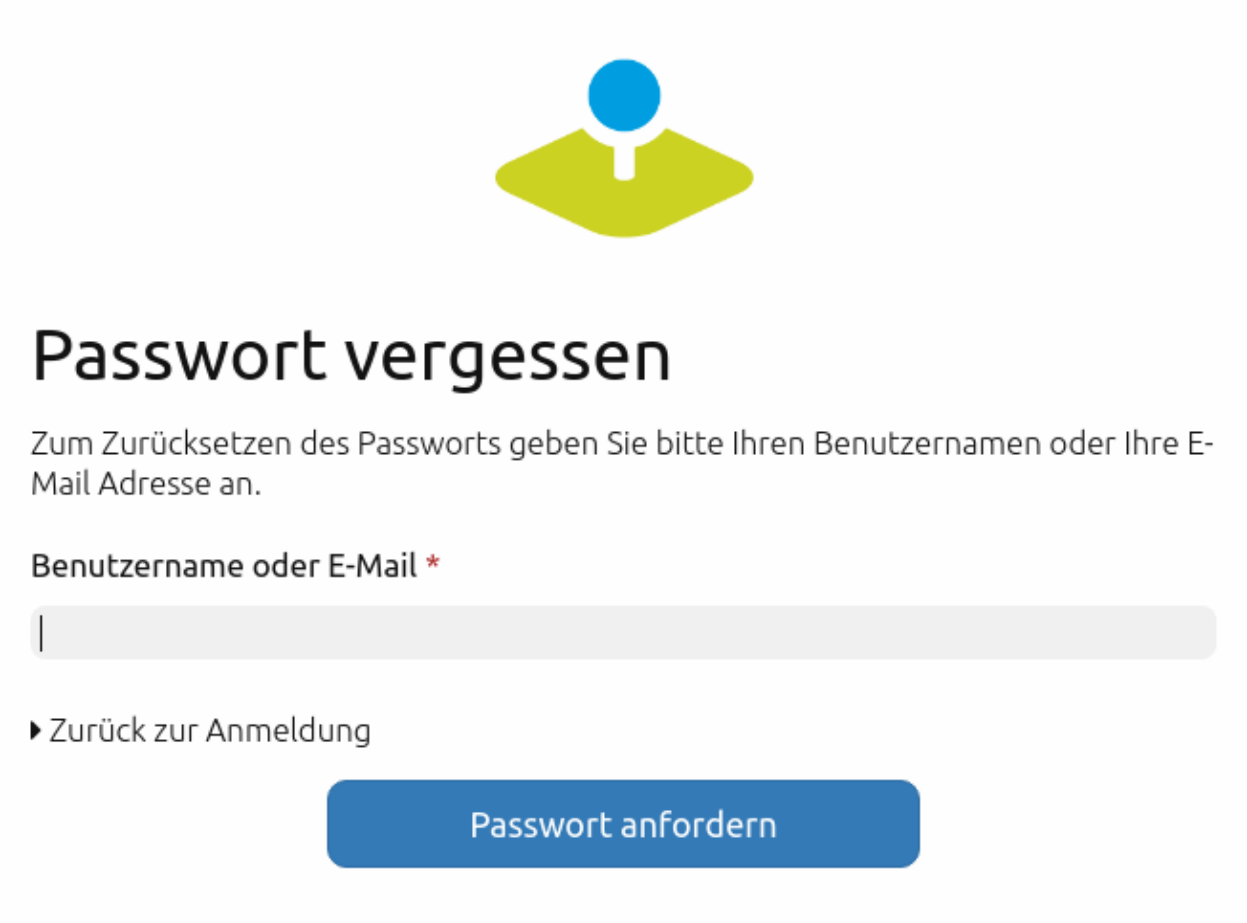
Das Bundle enthält Möglichkeiten für folgende Optionen:

- Benutzerverwaltung durch einen Administrator
- Registierung eines Benutzers
- Zurücksetzung des eigenen Passworts

Der Benutzer mit der ID 1 (root) ist besonders, da dieser Benutzer bei der Installation erstellt wird und immer alle Rechte hat. Falls alle Stricke reißen, können Sie mit diesem Benutzer alles verwalten. Und falls Sie gar die Anmeldedaten vergessen haben sollten, können Sie über ein *bin/console Kommando* den Benutzer zurücksetzen: `fom:user:resetroot`.

Passwort vergessen

Falls ein Benutzer sein Passwort vergessen hat, kann er in der Login-Maske über den Link “Passwort vergessen” ein neues Passwort anfordern. Dazu gibt er dann seinen Benutzernamen oder seine E-Mail Adresse an.



Danach bekommt der Benutzer eine E-Mail mit einem Link zur Zurücksetzung des Passworts. Der Link ist nach der Nutzung nicht mehr gültig. Der Text der Mail kann in der Datei */FOM/UserBundle/Resources/translations/messages.de.xlf* angepasst werden.

Die Funktionalität kann in der doctrine.yaml ausgeschaltet werden.

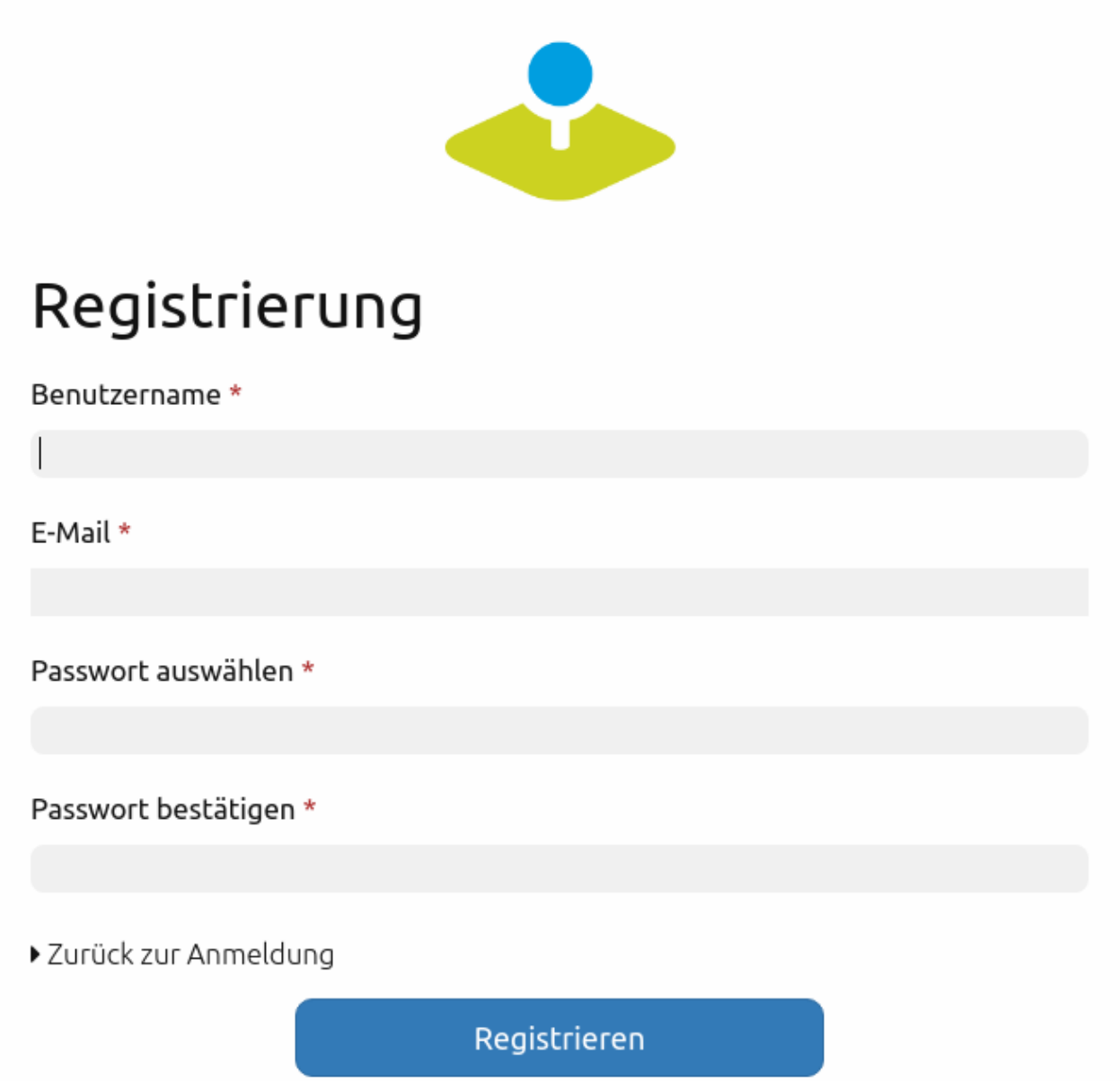
```
fom_user:
    reset_password: true # true/false
```

Registrierung

Benutzer können sich in Mapbender selbst registrieren. Vorher muss in der doctrine.yaml die Einstellung *fom_user:selfregister* auf true gestellt werden.

```
fom_user:
    selfregister: false # true/false
```

Im Login-Dialog erscheint der *Registrieren* Button. Der Benutzer wird zu einer Maske geführt, in der er Name, Passwort und E-Mail-Adresse angeben kann.

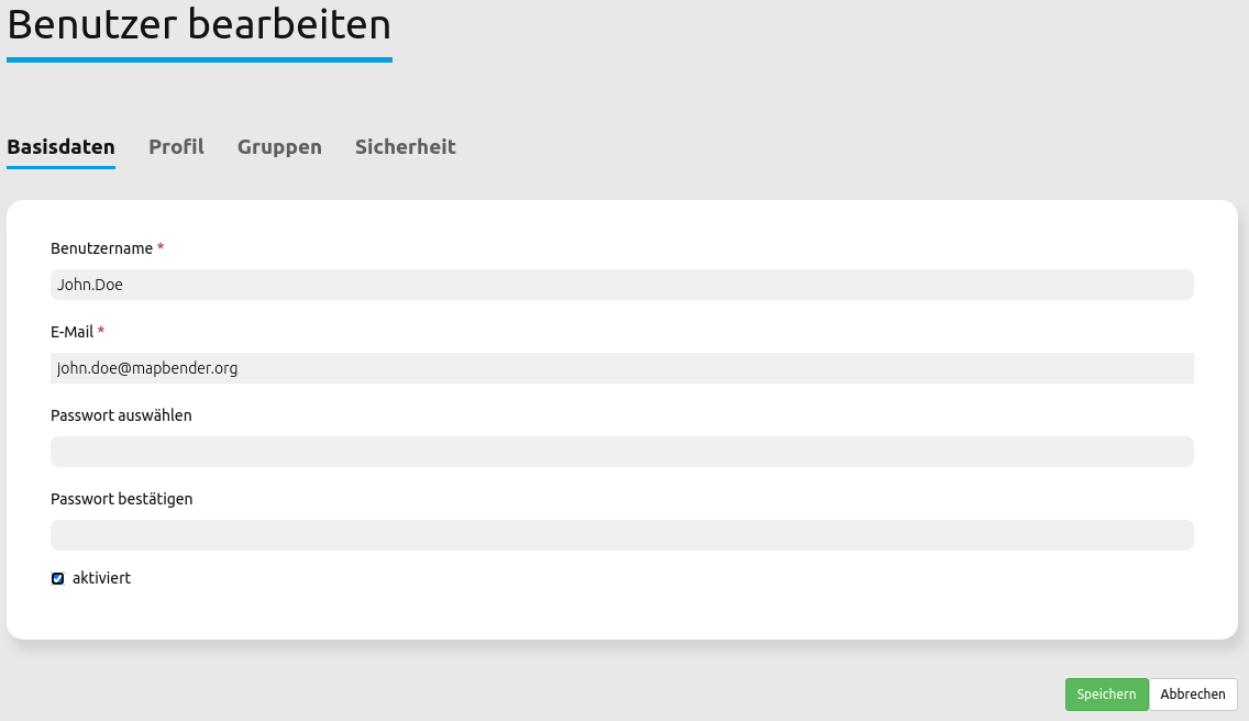


Danach erhält er eine Bestätigungsmail, mit der er seine Anmeldung abschließen kann. Bis zu diesem Zeitpunkt ist er als inaktiver Nutzer in Mapbender hinterlegt.

Die Texte der Bestätigungsmail können unter */FOM/UserBundle/Resources/translations/messages.de.xlf* angepasst werden.

Aktivieren von Nutzern

Benutzer können von Administratoren über eine entsprechende Checkbox aktiviert oder deaktiviert werden. Ein Benutzer mit Administrationsrechten kann sich selbst nicht aktivieren oder deaktivieren.



Ein Benutzer, der deaktiviert ist, kann sich so lange nicht mehr im Mapbender anmelden, bis er wieder aktiviert wird.

Benutzer, die sich selbst registriert haben, aber die Freischaltungsmail noch nicht bestätigt haben, können so von einem Administrator per Hand freigeschaltet werden.

Zuweisung der Rechteverwaltung an Benutzer oder Gruppen

1. Gehen Sie zu **Sicherheit Globale Berechtigungen**.
2. Wählen Sie **Rechteverwaltung**.
3. Weisen Sie Rechte an Benutzer oder Gruppen über die Schaltfläche **Füge Benutzer und Gruppen hinzu** zu. Diese erhalten das Recht *Globale Rechteverwaltung*.
4. Sie haben nun einem Benutzer/einer Gruppe die Kontrolle über andere Benutzerkonten zugewiesen.
5. Mit dieser Berechtigung können Benutzer und Gruppen anderen Benutzern und Gruppen Rechte über **Sicherheit Benutzer** oder **Sicherheit Gruppen** zuweisen.

Login Fehler

Fehlerhafte Logins werden mit der Meldung “Login fehlerhaft” kommentiert. Loginfehler schließen den Account nicht dauerhaft aus. Vielmehr wird der Account für eine bestimmte Zeit ausgeschlossen (gelockt).

Die doctrine.yaml ermöglicht die Anpassung dieses Verhaltens:

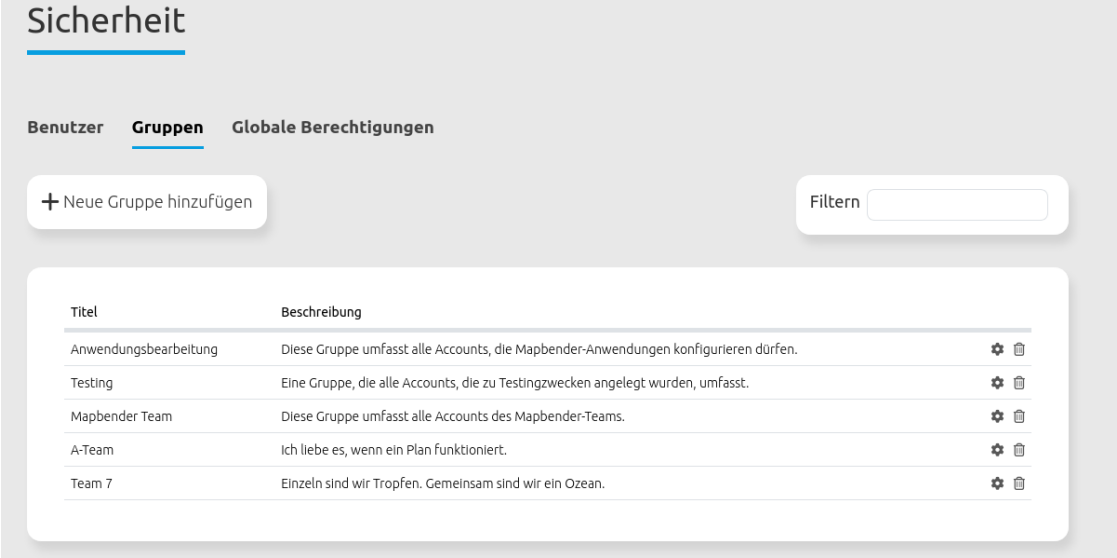
```
fom_user:

    # Allow to create user log table on the fly if the table doesn't exits.
    # Default: true
    auto_create_log_table: true

    # Time between to check login tries
    login_check_log_time: "-5 minutes"

    # Login attemps before delay starts
    login_attempts_before_delay: 3

    # Login delay after all attemps are failed
    login_delay_after_fail: 2 # Seconds
```

- **auto_create_log_table:** Angabe zur Rückwärtskompatibilität (Standard: true).
- **login_check_log_time:** Angabe zur Bereinigung der Login-Failure Tabelle (Standard: -5 minutes)
- **login_attempts_before_delay:** Anzahl der Login Versuche, bevor das Login-Delay greift (Standard: 3)
- **login_delay_after_fail:** Länge des Login-Delays in Sekunden (Standard: 2).

4.3.2 Gruppen und Rollen

Gruppen

Gruppen sind selbst erstellte Datenbank-Entitäten. Benutzer können ihnen individuell zugewiesen werden. Gemeinsame Gruppenmitglieder besitzen die gleichen Rechte, sofern diese global für die Gruppe konfiguriert wurden. Gruppen werden, die entsprechende Berechtigung vorausgesetzt, über `Sicherheit` → `Gruppen` → `Neue Gruppe hinzufügen` angelegt und konfiguriert. Dort bietet Mapbender ebenfalls eine Übersicht über alle Gruppen an:

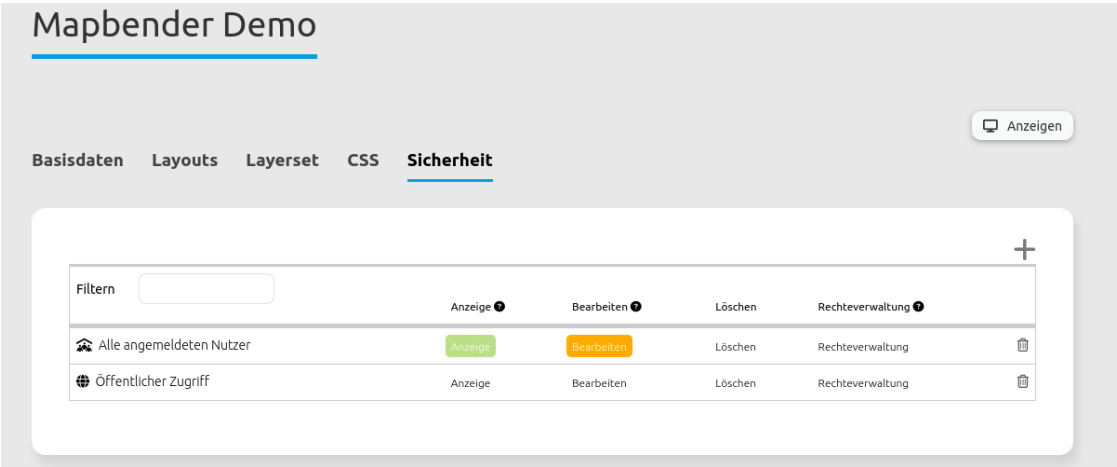


Rollen

Rollen sind Zugehörigkeiten zu systemeigenen Standard-Gruppen. Die folgenden Rollen sind für eine übergreifende Rechtevergabe auf bestimmte Domänen-Objekte verfügbar:

- **Alle angemeldeten Nutzer**: Rechtevergabe betrifft alle Benutzer mit einem Account.
- **Öffentlicher Zugriff**: Rechtevergabe betrifft alle unangemeldeten Besucher von Mapbender-URLs.

Rollen können je nach Kontext in der `Anwendungssicherheit` oder im Bereich `Globale Berechtigungen` Berechtigungen zugewiesen werden. Eine individuelle Anwendung könnte dann beispielsweise wie folgt abgesichert sein:



Hint: Globale Berechtigungen für einen Bereich weisen Sie über `Sicherheit` → `Globale Berechtigungen` zu.

4.3.3 Rechteverwaltung

Mapbender bietet zur Absicherung der Anwendungsbereiche unterschiedliche Stufen von Berechtigungen an, die Sie vergeben können.

- **Erstellen**: Ein neues Objekt erzeugen
- **Ansehen**: Ein existierendes Objekt lesen
- **Bearbeiten**: Ein existierendes Objekt editieren
- **Löschen**: Ein existierendes Objekt entfernen
- **Berechtigungen verwalten**: Ermöglicht die Bearbeitung von Berechtigungen
- **Aktualisieren**: Ermöglicht die Aktualisierung von Datenquellen

Fügen Sie Benutzer und Gruppen über `Sicherheit` → `Globale Berechtigungen` → `Rechteverwaltung` diesem Bereich hinzu, damit sie Zugriff auf das Rechteverwaltungssystem erhalten.

CHAPTER

FIVE

ANPASSUNGEN / TIPPS & TRICKS

5.1 Wie werden eigene Style-Vorlagen (templates) erzeugt?

Mapbender beinhaltet bereits Anwendungs-Vorlagen. Sie befinden sich im Mapbender CoreBundle Template-Verzeichnis */application/mapbender/src/Mapbender/CoreBundle/Template*. Häufig sollen jedoch eigene Anwendungs-Vorlagen und Administrationsoberflächen mit eigenem Corporate Design verwendet werden. Um Probleme bei einem Upgrade zu vermeiden, sollte für personalisierte Oberflächen ein eigenes Bundle verwendet werden. Der Stil einer einzelnen Anwendung kann ebenfalls über den *CSS-Editor* angepasst werden.

5.1.1 Wie werden eigene Vorlagen erzeugt?

Schritte für die Einbindung:

- Erzeugen eines eigenen Bundles,
- Erzeugen eines neuen Namespaces,
- Template PHP-Datei zur Registrierung der eigenen Vorlage erzeugen,
- Erzeugen einer eigenen Twig-Datei,
- Erzeugen eigener CSS-Datei(en),
- Verwenden der neuen Vorlage.

Als Beispiel für die Einbindung einer eigenen Vorlage wurde ein Workshop/DemoBundle vorbereitet. Dieses kann verwendet und individuell verändert werden. Sie können sich die Dateien im Workshop-Bundle herunterladen.

5.1.2 Erzeugen eines eigenen Bundles

Anwender-Bundles werden im *src*-Verzeichnis abgelegt. Die Struktur eines Bundles kann wie folgt aussehen:

```
src/Workshop/DemoBundle/

                        WorkshopDemoBundle.php

                        /Resources
                                  /public
                                         demo_fullscreen.css
```

(continues on next page)

(continued from previous page)

```
                                    /image
                                           workshop.ico
                                           workshop_logo.png
                                           print.png

                                    /views
                                                                /Template
                                                        fullscreen_demo.html.twig
                        /Template
                                            DemoFullscreen.php
```

Die folgenden Dateien können bearbeitet werden:

- twig-Datei: verändert die Struktur (z.B. - Löschen einer Komponente wie die Sidepane),
- css-Datei: verändert die Farben, Icons, Schriften.

5.1.3 Erzeugen eines neuen Namespaces

In unserem Beispiel erzeugt die Datei *WorkshopDemoBundle.php* einen Namespace für das Bundle und referenziert auf das Template.

```
<?php

namespace Workshop\DemoBundle;

use Mapbender\CoreBundle\Component\MapbenderBundle;

class WorkshopDemoBundle extends MapbenderBundle
{
    public function getElements()
    {
        return array(
        //    'Workshop\DemoBundle\Element\MapKlick',
        );
    }
    public function getTemplates()
    {
        return array('Workshop\DemoBundle\Template\DemoFullscreen');
    }
}
```

5.1.4 Anlegen der eigenen Template-Datei

In unserem Beispiel heißt die Template-Datei *DemoFullscreen.php*. Sie befindet sich unter *src/Workshop/DemoBundle/Template/DemoFullscreen.php*. In der Templates-Datei wird der Name des Templates, die Regionen die angelegt werden sollen sowie die verwendete Twig-Datei definiert.

```
<?php

namespace Workshop\DemoBundle\Template;

use Mapbender\CoreBundle\Template\Fullscreen;
```

(continues on next page)

(continued from previous page)

```
class DemoFullscreen extends Fullscreen
{
    protected static $title             = "Fullscreen Template Workshop";
    protected static $regions           = array('toolbar', 'sidepane', 'content', 'footer');
    protected static $regionsProperties = array(
        'sidepane' => array(
            'tabs'      => array(
                'name'  => 'tabs',
                'label' => 'mb.manager.template.region.tabs.label'),
            'accordion' => array(
                'name'  => 'accordion',
                'label' => 'mb.manager.template.region.accordion.label')
        )
    );
    protected static $css               = array(
        '@MapbenderCoreBundle/Resources/public/sass/template/fullscreen.scss',
        '@WorkshopDemoBundle/Resources/public/demo_fullscreen.scss',
    );
    protected static $js                = array(
        '@FOMCoreBundle/Resources/public/js/frontend/sidepane.js',
        '@FOMCoreBundle/Resources/public/js/frontend/tabcontainer.js',
        '@MapbenderCoreBundle/Resources/public/mapbender.container.info.js',
        '/components/jquerydialogextendjs/jquerydialogextendjs-built.js',
        "/components/vis-ui.js/vis-ui.js-built.js"
    );

    public function getTwigTemplate()
    {
        return '@WorkshopDemoBundle/Resources/views/Template/demo_fullscreen.html.twig';
    }
```

5.1.5 Eigene Twig-Datei erzeugen

Die Twig-Dateien sind im folgenden Verzeichnis gespeichert:

- *application/mapbender/src/Mapbender/CoreBundle/Resources/views/Template*

Kopieren Sie eine existierende Twig-Datei, speichern Sie diese unter einem neuen Namen und verändern Sie den Inhalt, z.B. die Farbe.

Verwenden Sie z.B. *mapbender/src/Mapbender/CoreBundle/Resources/views/Template/fullscreen.html.twig* und kopieren Sie diese nach */Workshop/DemoBundle/Resources/views/Template/fullscreen_demo.html.twig*.

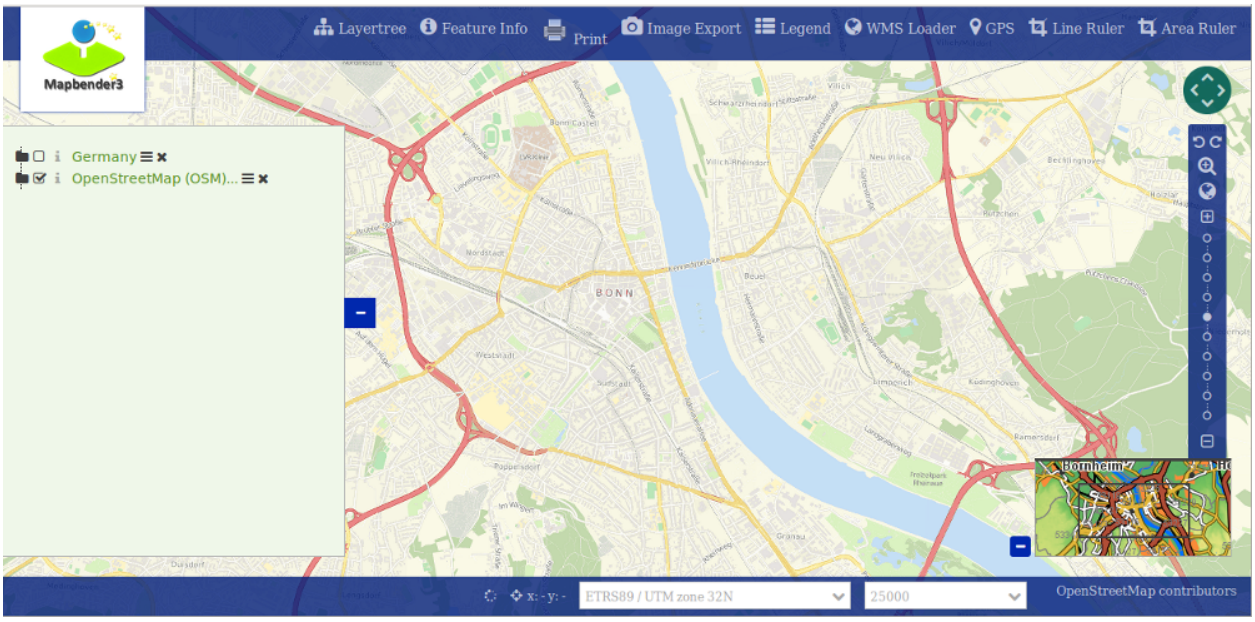
5.1.6 Eigene CSS-Datei erzeugen

Erzeugen Sie eine leere CSS-Datei und geben Sie in dieser nur die CSS-Definitionen für Ihre Anwendungs-Vorlage an. Es muss lediglich das CSS definiert werden, das vom Standard der Elemente abweicht. Mit Hilfe der Entwicklerwerkzeuge Ihres Browsers können Sie die bestehende Definition ermitteln, in Ihre CSS-Datei kopieren und hier anpassen.

Ihre CSS-Datei könnte wie folgt heißen: *src/Workshop/DemoBundle/Resources/public/demo_fullscreen.css* und die folgende Definition enthalten:

```
.toolBar {
  background-color: rgba(0, 29, 122, 0.8) !important;
}

.toolPane {
  background-color: rgba(0, 29, 122, 0.8) !important;
}

.sidePane {
  overflow: visible;
  background-image: url("");
  background-color: #eff7e9;
}

.sidePane.opened {
    width: 350px;
}

.logoContainer {
  background-color: white !important;
  background-image: url("") !important;
  -webkit-box-shadow: 0px 0px 3px #0028AD;
  -moz-box-shadow: 0px 0px 3px #0028AD;
  box-shadow: 0px 0px 3px #0028AD;
}

.sidePaneTabItem {
   background-color: #0028AD;
}

.layer-opacity-handle {
    background-color: #0028AD;
}

.mb-element-overview .toggleOverview {
    background-color: #0028AD;
}

.button, .tabContainerAlt .tab {
    background-color: #0028AD;
}

.iconPrint:before {
  /*content: "\f02f"; }*/
  content:url("image/print.png");
}

.popup {
  background-color: #eff7e9;
  background-image: url("");
}

.pan{
  background-color: rgba(0, 93, 83, 0.9);
}
```

Das Ergebnis der wenigen Zeilen CSS sieht dann so aus:



Beim Laden der neuen Anwendung wird eine CSS-Datei im web/assets-Verzeichnis angelegt:

- *web/assets/WorkshopDemoBundle__demo_fullscreen__css.css*

Wenn Sie die CSS-Datei weiter bearbeiten, müssen Sie die unter `web/assets` generierte Datei löschen, damit diese neu geschrieben wird und die Änderungen wirksam werden. Der Browser-Cache sollte ebenfalls geleert werden.

```
sudo rm -f web/assets/WorkshopDemoBundle__demo_fullscreen__css.css
```

5.1.7 Styling der Administrationsseiten

Passen Sie die vorhandenen CSS-Dateivorlagen für die unterschiedlichen Bereiche an:

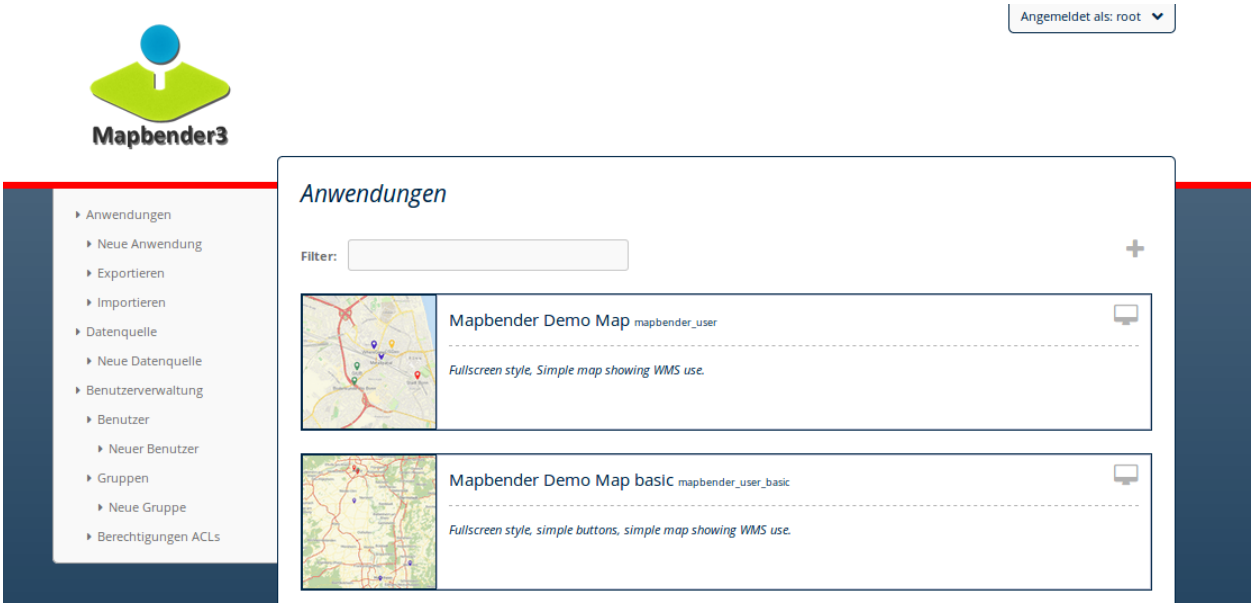
- *login.css*: Anpassung des Designs der Login-Oberfläche (**Anmelden**-Seite),
- *manager.css*: Anpassung des Designs der Verwaltungs/Administrations-Oberfläche (**Anwendungsübersicht** etc.),
- *password.css*: Anpassung des Designs der Passwort-Oberfläche (**Passwort vergessen**).

Es muss lediglich das CSS definiert werden, das vom Standard der Administrationsoberfläche abweicht.

Auf die CSS-Dateien wird über das FOMManagerBundle und FOMUserBundle referenziert. Diese müssen unter *app/Resources/* abgelegt werden. Die bereits enthaltenen Twig-Dateien überschreiben nach der erfolgreichen Einrichtung die Standard-Einstellungen (Vorgaben aus der *manager.html.twig* Datei). Alternativ kann auch die bisherige Twig-Datei kopiert und angepasst werden.

```
cp fom/src/FOM/ManagerBundle/Resources/views/manager.html.twig app/Resources/FOMManagerBundle/views/
```

Bei unveränderter Übernahme der Stylevorgaben sieht die Administration dann so aus:



5.1.8 Registrieren des Bundles in config/bundles.php

Falls noch nicht geschehen, setzen Sie Schreibrechte für das *public*-Verzeichnis für Ihren Webserver-Benutzer:

```
chmod ug+w public
```

Aktualisieren Sie das *public*-Verzeichnis. Jedes Bundle hat seine eigenen Assets - CSS Dateien, JavaScript-Dateien, Bilder und mehr - diese müssen in das *public*-Verzeichnis kopiert werden.

```
bin/console assets:install public
```

Mit der Option `--symlink --relative` werden die Dateien nicht kopiert. Es wird stattdessen ein symbolischer Link erzeugt. Dies erleichtert das Editieren innerhalb des Bundles.

```
bin/console assets:install public --symlink --relative
```

Jetzt sollte beim Anlegen einer neuen Anwendung die neue Vorlage in der Liste erscheinen.

5.1.9 Verwendung der eigenen Vorlage

Die neue Anwendungs-Vorlage kann über verschiedene Wege verwendet werden:

Einbindung in YAML-Anwendungen

Sie können nun die YAML-Anwendungen unter *config/applications* anpassen und auf das neue Template verweisen.

```
template: Workshop\DemoBundle\Template\DemoFullscreen
```

Einbindung in eine neue Anwendungen über das Backend

Wenn Sie eine neue Anwendung mit der Mapbender-Administration erzeugen, können Sie eine Vorlage (Template) auswählen.

Einbindung in eine bestehende Anwendung

Für bereits existierende Anwendungen kann das Template über die Mapbender Datenbank in der Tabelle **mb_core_application** in der Spalte **template** angepasst werden.

Für das **WorkshopDemoBundle** wird hier statt des Eintrags `Mapbender\CoreBundle\Template\Fullscreen` der Eintrag `Workshop\DemoBundle\WorkshopDemoBundle` angegeben.

5.1.10 Anwendungsfälle

F: Wie kann das Logo verändert werden? Wie kann der Anwendungstitel angepasst werden? Wie stelle ich die Sprache ein?

A: Auf diese und weitere Fragen finden Sie Antworten im Kapitel *YAML Konfiguration (Konfigurations- und Anwendungsdateien)*.

5.1.11 Wie können eigene Buttons eingebunden werden?

Mapbender verwendet Schrift-Icons auf der FontAwesome Collection:

```
@font-face {
  font-family: 'FontAwesome';
  src: url("../../bundles/fomcore/images/icons/fontawesome-webfont.eot?v=3.0.1");
  src: url("../../bundles/fomcore/images/icons/fontawesome-webfont.eot?#iefix&v=3.0.1") format(
↪"embedded-opentype"), url("../../bundles/fomcore/images/icons/fontawesome-webfont.woff?v=3.0.1")
↪format("woff"), url("../../bundles/fomcore/images/icons/fontawesome-webfont.ttf?v=3.0.1") format(
↪"truetype");
  font-weight: normal;
  font-style: normal;
}
```

In der CSS-Datei können Sie zu den Icons der Schriftart folgendermaßen verweisen:

```
.iconPrint:before {
  content: "\f02f";
}
```

Wenn Sie ein Bild nutzen möchten, legen Sie dieses am besten in Ihrem Bundle ab und referenzieren es auf die folgende Art und Weise:

```
.iconPrint:before {
  content:url("image/print.png");
}
```

5.1.12 Probieren Sie es aus

- Laden Sie das Workshop/DemoBundle hier herunter.
- Ändern Sie die Farbe Ihrer Icons.
- Ändern Sie die Größe Ihrer Icons.
- Ändern Sie die Farbe der Oberen Werkzeugleiste.
- Benutzen sie ein Bild anstelle eines Font-Icons für Ihre Buttons.
- Verschieben Sie die Übersichtskarte auf die linke Seite.
- Schauen Sie in die Workshop-Dateien, um zu erfahren, wie das funktioniert.

5.2 YAML Konfiguration (Konfigurations- und Anwendungsdateien)

Die folgenden Konfigurationsdateien liegen unter *config/* und dessen Unterverzeichnissen:

5.2.1 doctrine.yaml

Diese Datei enthält grundlegende Architektur-Vorgaben von Mapbender. Gleichzeitig sind hier die Parameter für die *parameters.yaml* als Platzhalter definiert. Des Weiteren legt die Datei fest, welche Konfigurationen für den produktiven Modus und den Entwicklungsmodus verwendet werden sollen.

- **fom_user.selfregistration**: Um die Selbstregistrierung zu de/aktivieren, passen Sie diesen Parameter an. Sie müssen unter self_registration_groups eine/mehrere Gruppen angeben, so dass selbstregistriere Anwender automatisch (bei der Registrierung) diesen Gruppen zugewiesen werden. Über die Gruppe bekommen Sie dann entsprechend Rechte zugewiesen.
- **fom_user.reset_password**: Über diesen Parameter kann die Möglichkeit de/aktiviert werden, das Passwort neu zu setzen.
- **framework.session.cookie_httponly**: Stellen Sie für HTTP-only session cookies sicher, dass der Parameter framework.session.cookie_httponly auf true steht.

Datenbank-Konfiguration

Zur Konfiguration der Datenbankverbindung erfolgt in den Dateien `.env.local` und `doctrine.yaml`. In der `.env.local` wird die Datenbankverbindung definiert in einer Variable. Die Variablen werden in der `doctrine.yaml` angegeben.

Note: Jede Datenbank, die in der *.env* definiert wird, kann in der *doctrine.yaml* unter url zu einer Verbindung angegeben werden:

```
doctrine:
    dbal:
        default_connection: default                    # gibt die Datenbankverbindung an, die
↪standardmäßig von Mapbender verwendet werden soll (``default_connection: default``).
        connections:
            default:
            url: '%env(resolve:MAPBENDER_DATABASE_URL)%'# Platzhalter, der auf die definierte
↪Umgebungsvariable in der parameters.yaml verweist.
            persistent: true                           # Parameter, ob die Verbindung zur Datenbank
↪dauerhaft hergestellt werden soll.
            charset:    UTF8                           # Die Kodierung, die die Datenbank verwendet.
            logging:   "%kernel.debug%"                # Die Option sorgt dafür, das alle SQLs nicht
↪mehr geloggt werden (Standard: %kernel.debug%). `Mehr Informationen <http://www.loremipsum.at/blog/
↪doctrine-2-sql-profiler-in-debugleiste>`_.
            profiling: "%kernel.debug%"                # Profiling von SQL Anfragen. Diese Option kann
↪in der Produktion ausgeschaltet werden. (Standard: %kernel.debug%)
            #server_version: '15'                      # Wichtig: Sie MÜSSEN die Serverversion
↪konfigurieren, entweder hier oder in der DATABASE_URL Umgebungsvariable (siehe .env-Datei).
```

Verwendung mehrerer Datenbanken

Es folgt ein Beispiel mit zwei Datenbankverbindungen in der *doctrine.yaml*:

```
doctrine:
    dbal:
        default_connection: default
        connections:
            # Datenbankverbindung default
            default:
                url: '%env(resolve:MAPBENDER_DATABASE_URL)%'
                charset:    UTF8
                #server_version: '15'
                logging:   "%kernel.debug%"
                profiling: "%kernel.debug%"
            # Datenbankverbindung geodata_db
            geodata_db:
                url: '%env(resolve:GEOBASIS_DATABASE_URL)%'
                persistent: true
                charset:  UTF8
                logging: '%kernel.debug%'
                profiling: '%kernel.debug%'
                # IMPORTANT: You MUST configure your server version,
                # either here or in the DATABASE_URL env var (see .env file)
                #server_version: '15'
```

5.2.2 .env bzw. .env.local

In dieser Datei werden zentrale Umgebungsvariablen zusammengeführt:

Datenbank

Zur Konfiguration der Datenbankverbindung werden die Dateien *.env* und *doctrine.yaml* verwendet. In der *.env* wird die Umgebungsvariable für die Datenbankverbindung definiert. *.env* wird von *.env.local* überschrieben. Diese wird in der *doctrine.yaml* ausgelesen. In der *doctrine.yaml* können auch mehrere Datenbankverbindungen definiert werden. Zu jeder Datenbankverbindung wird ein Alias vergeben.

Beispiel: Die Datenbankkonfiguration in der *.env* sieht standardmäßig folgendermaßen aus, sofern die mitinstallierte SQLite-Datenbank verwendet wird:

```
MAPBENDER_DATABASE_URL="sqlite:///%kernel.project_dir%/var/db/demo.sqlite"
```

Beispiel: Die Datenbankkonfiguration in der *.env.local* sieht wie folgt aus, wenn eine Suchdatenbank via PostgreSQL referenziert wird:

```
SEARCH_DB_DATABASE_URL="postgresql://dbuser:dbpassword@localhost:5432/dbname?serverVersion=14&
↪charset=utf8"
```

Verwendung mehrerer Datenbanken

Mit Mapbender können Sie auch mehrere Datenbanken verwenden. Dies wird bei der Einbindung von Geodaten empfohlen. Diese sollten getrennt von der Mapbender-Datenbank liegen. Eine zweite Datenbank benötigen Sie ebenfalls für die Geodatensuche (über den *Suchen (Search Router)*) und die Datenerfassung (*Digitizer*). Die Standard-Datenbankverbindung (`default_connection: default`) wird von Mapbender verwendet.

Wenn Sie eine weitere Datenbank verwenden möchten, müssen Sie eine zweite Datenbankverbindung mit einem anderen Namen in der *.env.local*-Datei definieren. In den Elementen *Suchen (Search Router)* und *Digitizer* kann nun auf die Datenbankverbindung (connection) mit dem Namen *search_db* verwiesen werden.

Weitere Information über diese Konfigurationsmöglichkeit gibt es in der Symfony Dokumentation. Mapbender verwendet Doctrine. Doctrine ist eine Sammlung von PHP-Bibliotheken und bietet einen objektrelationalen Mapper und eine Datenbankabstraktionsschicht (Doctrine Projektseite).

Mailer

Die Angabe zum Mailer wird in der *.env.local* Datei über die Variable `MAILER_DSN` definiert.

```
#MAILER_DSN=smtp://user:pass@smtp.example.com:25
MAILER_DSN=null://null
```

Der Mailer-Einstellugnen selbst werden in *fom.yaml* konfiguriert.

```
fom_user:
    selfregister: false
    reset_password: true
    max_reset_time: 1
    mail_from_address: info@mapbender.org
    mail_from_name: Mapbender Team
```

Hint: Ein Mailer wird für die Funktionen 'Registrierung' und 'Passwort zurücksetzen' benötigt. Weitere Informationen im Kapitel *Benutzer*.

5.2.3 parameters.yaml

Hier werden weitere grundlegende Parameter von Mapbender bestimmt.

Disclaimer

Es kann ein Disclaimer mittels Sitelinks hinzugefügt werden. Dafür muss Folgendes in der *parameters.yaml* ergänzt werden:

```
mapbender.sitelinks:
  - link: https://mapbender.org/impressum                    # Link URL
    text: Impressum & Kontakt                                                #
↪Link Text
  - link: https://mapbender.org/datenschutz
    text: Datenschutz
  - link: https://doc.mapbender.org/
    text: Dokumentation
```

Die Sitelinks werden mittels "|" voneinander getrennt.

Elemente deaktivieren

Die Deaktivierung von einzelnen Elementen kann über den nachfolgenden Parameter konfiguriert werden:

```
mapbender.disabled_elements:
```

Icons anpassen

Mithilfe des Setzens folgender Parameter wird die spezifische Anpassung von Icons ermöglicht:

1. `mapbender.icons.disable_default` (Standardwert: false).
   - Wenn auf *true* gesetzt, werden die mitgelieferten Icons des Mapbender-Namespace im Button-Bearbeitungsformular deaktiviert. Dazu gehören die Icons About, Area Ruler, Feature Info, Image Export, Layer tree, Legend, Line ruler und Polygon. Diese werden z.B. für die ähnlich bezeichneten Elemente *Über-Mapbender-Dialog (About Dialog)*, *Linien-/Flächenmessung (Line/Area Ruler)*, *Information (FeatureInfo)*, *Bildexport (ImageExport)*, *Ebenenbaum (Layertree)*, *Legende (Legend)*, *Digitizer* und *Skizzen (Sketch)* verwendet.
2. `mapbender.icons.disable_default_fa` (Standardwert: false).

- Wenn auf *true* gesetzt, werden die mitgelieferten Font Awesome-Icons im Button-Bearbeitungsformular deaktiviert. Dies betrifft alle anderen verfügbaren Icons, die nicht im Mapbender-Namespace enthalten sind.

3. `mapbender.icons.custom` (Standardwert: ~).
   - Ermöglicht das Hinzufügen weiterer Icons aus FontAwesome (eine Liste frei verfügbarer Icons finden Sie auf der Font Awesome-Homepage.
   - Der Parameter ist ein Array von Objekten, die drei Schlüssel enthalten:
     - *name*: Bezeichnung für dieses Icon, z.B. zur Verwendung in yaml-definierten Anwendungen.
     - *title*: Beschreibung des Icons, wie es im Auswahlfeld im Bearbeitungsformular für Schaltflächen angezeigt wird.
     - *class*: Vollständiger Klassenname (einschließlich "fa") des Icons. Kopieren Sie ihn aus dem unteren Teil des Popups, wenn Sie auf das Icon auf der Font Awesome-Homepage klicken.

Mit diesen Konfigurationsoptionen können Sie die Icons in Mapbender an Ihre Anforderungen anpassen. Anbei ein Beispiel:

```
mapbender.icons.disable_default: true
mapbender.icons.disable_default_fa: true
mapbender.icons.custom:
    - name: iconSnowflake
      title: Snowflake
      class: fa-solid fa-snowflake
    - name: iconHeartbeat
      title: Heartbeat
      class: fa-solid fa-heart-pulse
```

Logo und Login-Bild

In der *parameters.yaml* kann auf das eigene Logo und auf ein alternatives Bild für den Login verwiesen werden. Diese Änderung wirkt sich global auf die gesamte Mapbender-Installation aus.

Mithilfe der Branding-Parameter kann ein eigener Projektname, eine eigene Versionsnummer, ein Logo, eine eigene Favoriten-Symbolgrafik und ein alternatives Bild für den Login verwendet werden. Diese Änderung wirkt sich global auf die gesamte Mapbender-Installation aus.

```
branding.project_name: Geoportal powered by Mapbender
branding.project_version: 1.0
branding.logo: ./bundles/mapbendercore/image/OSGeo_project.png
branding.favicon: ./application/public/brand-favicon.ico
branding.login_backdrop: ./bundles/mapbendercore/image/login-backdrop.jpg
```

Die Dateien müssen unter `application/public` verfügbar sein.

Projektname

Der Projektname (Standard: Mapbender) kann in der Datei *parameters.yaml* angepasst werden. Diese Änderung wirkt sich global auf die gesamte Mapbender Installation aus.

```
branding.project_name: Geoportal
```

Warning: In der *parameters.yaml* dürfen **keine Tabulatoren für Einrückungen** verwendet werden. Nutzen Sie stattdessen Leerzeichen.

Proxy-Einstellungen

Wenn ein Proxy verwendet wird, muss dieser in der Datei *parameters.yaml* im Bereich OWSProxy Configuration angegeben werden.

Eine Konfiguration könnte wie folgt aussehen:

```
# OWSProxy Configuration
    ows_proxy3_logging: false             # Protokollierung von Anfragen, Standard ist false, true
↪protokolliert in Tabelle owsproxy_log
    ows_proxy3_obfuscate_client_ip: true  # Verbergen der Client IP, Standard ist true, true verbirgt
↪das letzte Byte der IP-Adresse des Clients
    ows_proxy3_host: myproxy              # Proxy-Definition für die Verbindung über einen Proxy-Server.
↪ Hostname des Proxyservers
    ows_proxy3_port: 8080                 # Proxy-Definition für die Verbindung über einen Proxy-Server.
↪ Port des Proxyservers
    ows_proxy3_connecttimeout: 60
    ows_proxy3_timeout: 90
    ows_proxy3_user: ~                    # Benutzername für Proxyserver (bei Bedarf Benutzer für
↪Proxyserver festlegen)
    ows_proxy3_password: ~                # Passwort für den Proxy-Server (setzen Sie das Passwort für
↪den Proxy-Server, falls definiert)
    ows_proxy3_noproxy:                   # Liste der Hosts, bei denen die Verbindungen nicht über den
↪Proxyserver erfolgen soll
        - 192.168.1.123
```

Spracheinstellung

Mapbender verwendet automatisch die ausgewählte Sprache der Browsereinstellungen. Es ist jedoch möglich, eine bevorzugte Sprache (*fallback_locale*) zu definieren, die Mapbender bei unvollständigen Übersetzungen anstelle der Browsersprache nutzt.

Die Sprache kann nur für die gesamte Mapbender Installation angepasst werden (nicht für einzelne Anwendungen).

Folgende Sprachcodes sind verfügbar:

- en für Englisch (Standard)
- de für Deutsch
- es für Spanisch
- fr für französisch,
- it für Italienisch

- nl für Niederländisch
- pt für Portugiesisch
- ru für Russisch
- tr für Türkisch
- uk für Ukrainisch

Eine Konfiguration könnte wie folgt aussehen:

```
fallback_locale:    en
locale:             de
secret:             ThisTokenIsNotSoSecretChangeIt
```

Mapbender kann auch explizit eine Sprache verwenden. Dazu muss der Parameter `mapbender.automatic_locale: false` gesetzt werden. Anschließend nutzt Mapbender die unter locale definierte Spracheinstellung.

```
mapbender.automatic_locale: false
fallback_locale:    en
locale:             es
secret:             ThisTokenIsNotSoSecretChangeIt
```

Weitere Informationen unter translation.

SSL Zertifikat

Für Produktivumgebungen ist die Installation eines SSL-Zertifikats wichtig. Anschließend muss die Variable `parameters.cookie_secure` in Ihrer *parameters.yaml* auf `true` gesetzt werden. Dadurch wird sichergestellt, dass das Login-Cookie nur über sichere Verbindungen übertragen wird.

Überschreiben von JavaScript- und CSS/Sass-Ressourcen

Um genannte Ressourcen manuell zu überschreiben, können Sie als Alternative *zum Überschreiben im Bundle selbst* in Ihrer `paramaters.yaml`-Datei Folgendes hinzufügen:

```
mapbender.asset_overrides:
    "@MapbenderCoreBundle/Resources/public/sass/element/featureinfo.scss": "@@MyBundle/Resources/public/
↪sass/element/custom_featureinfo.scss"
```

Note: Beachten Sie, dass das @-Zeichen im Ersetzungsschlüssel durch ein weiteres @@-Zeichen maskiert werden muss.

5.2.4 YAML Anwendungsdateien

Als YAML definierte Anwendungen werden im Verzeichnis *applications/* abgelegt. Dort liegen nach jeder Installation bereits drei Beispielanwendungen als YAML Dateien:

- Mapbender Demo (*mapbender_user*)
- Mapbender Demo Basic (*mapbender_user_basic*)
- Mapbender Mobile Demo (*mapbender_mobile*)

Falls eine Beispielanwendung nicht im Mapbender sichtbar sein soll, kann sie mit einem Texteditor geöffnet und die Variable `published` wie folgt angepasst werden:

```
parameters:
        applications:
                mapbender_mobile:
                        [...]
                        published: false
```

Löschen Sie anschließend den *Mapbender-Cache* und Ihren Browser-Cache. Danach ist die Anwendung für Benutzer nicht mehr sichtbar.

Hint: Der Root-Benutzer sieht immer alle unveröffentlichten Anwendungen.

Sie können außerdem die Anwendungsdateien aus dem Verzeichnis `applications` entfernen, um Sie aus der Mapbender-Instanz zu entfernen. Wiederholen Sie anschließend das Löschen der Caches.

Auf die gleiche Weise können auch neue YAML-basierte Anwendungen in dieses Verzeichnis abgelegt werden, um sie der Mapbender-Instanz hinzuzufügen.

5.2.5 Mapbender Demo Map

Dies ist die Demo-Anwendung, die für eine Desktop-Anwendung standardmäßig verwendet werden sollte.

Detaillierte Beschreibungen zu den enthaltenen Elementen finden Sie unter *Elemente*.

5.2.6 Mapbender Demo Map basic

Die zweite Demo-Anwendung, welche folgende Unterschiede zur ersten Demo-Anwendung aufweist:

Werkzeugleiste Enthält Elemente aus dem Seitenbereich der ersten Demo-Anwendung.

Seitenbereich Enthält keine im Voraus konfigurierten Elemente.

Kartenbereich Verwendet eine kompaktere *Navigationswerkzeug (Navigation Toolbar)* ohne den Zoom-Schieberegler. Beinhaltet die *Einfache Suche (SimpleSearch)*.

Detaillierte Beschreibungen der Elemente finden Sie unter *Elemente*.

5.2.7 Mapbender mobile

Diese Anwendung dient als mobile Vorlage für Smartphones und Tablets.

5.2.8 Export/Import von YAML Anwendungsdateien über die Benutzeroberfläche

Export

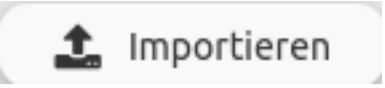
Sie können eine Anwendung unter **Anwendungen** **Exportieren** als JSON-Datei exportieren.

Nutzen Sie dazu den Exportieren-Button, der sich in der Anwendungsübersicht im Button-Menü einer Anwendung befindet.

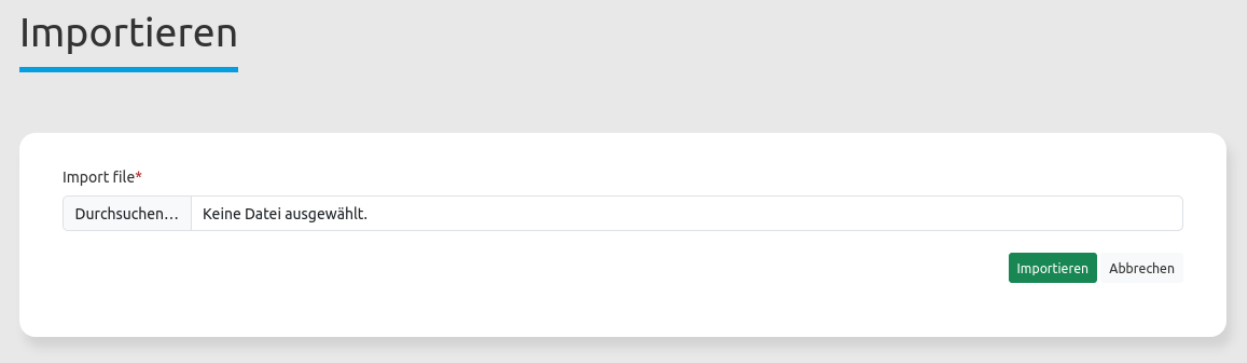
Import

Unter **Anwendungen** **Importieren** kann eine Exportdatei in eine Mapbender-Installation importiert werden.

Wählen Sie dazu zunächst den Button `+ Anwendung anlegen`. Anschließend klicken Sie auf den Importieren-Button.



Nutzen Sie danach die abgebildete Maske, um eine Importdatei als Anwendung zu laden.



5.2.9 Export/Import/Klonen von YAML Anwendungsdateien über die Konsole

Bitte gehen Sie zu *Anwendungs-Export, Import und Klonen*, um entsprechende Konsolenbefehle einzusehen. Nachfolgend finden Sie einige einführende Worte darüber, was mit Anwendungen über die Konsole möglich ist.

Export über die Konsole

Anwendungen können als .json oder .yaml - Datei über die Konsole exportiert werden. Jedoch kann eine YAML-Datei, die über die Konsole exportiert wurde, nicht unter *applications/* abgelegt und somit als Anwendung in Mapbender eingefügt werden. Das YAML-Format einer Datei, die über die Konsole exportiert wurde, unterscheidet sich von dem YAML-Format der Dateien unter *applications/*.

Import über die Konsole

YAML-Dateien, die zuvor über die Benutzeroberfläche oder die Konsole exportiert wurden, können über die Konsole via bin/console importiert werden.

Anwendung über die Konsole klonen

Klont/Kopiert eine existierende Anwendung.

5.3 bin/console Befehle

Mapbender bietet Befehle, die über die Kommandozeile aufgerufen werden können. Einige der Befehle werden von den Symfony-Komponenten bereitgestellt, andere gehören zu Mapbender.

Im Folgenden werden einige dieser Befehle vorgestellt.

Achten Sie beim Ausführen der Befehle darauf, dass Sie sich im richtigen Verzeichnis befinden (oberhalb vom app-Verzeichnis)

- mapbender/application (bei Installation über GitHub)
- mapbender (bei der Paketinstallation)

5.3.1 Hilfe zu den Befehlen

Die Hilfe für jeden Befehl kann mit dem `--help` Parameter aufgerufen werden, z. B.:

```
bin/console mapbender:user:create --help
```

5.3.2 Anwendungs-Export, Import und Klonen

bin/console mapbender:application:export

Eine Anwendung kann als JSON- oder YAML-Datei exportiert werden. Dabei muss der Anwendungs-URL-Titel (slug) angebeben und eine Exportdatei definiert werden.

Für .json-Dateien (Export über den Browser möglich):

```
bin/console mapbender:application:export mapbender_user_db --format=json > export.json
```

Für .yaml-Dateien (Export nur über die Kommandozeile möglich):

```
bin/console mapbender:application:export mapbender_user_db --format=yaml > export.yaml
```

bin/console mapbender:application:import

Eine Anwendung kann aus einer JSON- oder YAML-Datei importiert werden. Mapbender wählt automatisch einen neuen Namen, wenn der Name bereits vorliegt.

```
bin/console mapbender:application:import export.json

 Imported 1 applications
 * mapbender_user_db_imp2
```

bin/console mapbender:application:clone

Sie können auch eine bestehende Anwendung im Anwendungs-Backend klonen. Dadurch wird eine neue Anwendung mit einem Suffix im Anwendungsnamen erzeugt. Bei yaml-Anwendungen wird ein *_db*-Suffix, bei Anwendungen aus der Datenbank ein *_imp*-Suffix angehängt.

```
bin/console mapbender:application:clone mapbender_user_yml
```

5.3.3 Benutzerverwaltung

bin/console mapbender:user:create

Befehl zum Anlegen eines Benutzers über die Kommandozeile. Dabei sind die Angabe von Benutzername, Email und Passwort erforderlich. Der Benutzername und die Email müssen eindeutig sein.

```
bin/console mapbender:user:create --help
bin/console mapbender:user:create --password <password> --email <email> <name>
bin/console mapbender:user:create --password mypassword123 --email max.mustermann@mapbender.org 'Max
↪Mustermann'
```

Aktualisierung eines Benutzers

Die folgenden Angaben zu einem Benutzer können aktualisiert werden:

- E-Mail
- Passwort

Der Benutzername kann nicht verändert werden.

```
bin/console mapbender:user:create --update --password <password> --email <email> <name>

bin/console mapbender:user:create --update --password mypassword8910 --email max.mustermann@mapbender.
↪org 'Max Mustermann'
```

bin/console fom:user:resetroot

Mit diesem Befehl lässt sich der root-Account erstellen oder aktualisieren. Für die Erstellung müssen ein Benutzername, eine E-Mail und ein Passwort vergeben werden.

Bei der Aktualisierung erfolgt die eindeutige Zuordnung über die bereits vorhandene ID, deshalb können alle drei o.g. Parameter verändert werden.

```
bin/console fom:user:resetroot
```

```
bin/console fom:user:resetroot --username="root" --password="root" --email="root@example.com"
```

bin/console mapbender:user:list

Dieser Befehl zeigt im Terminal alle vorhandenen Benutzer mit ihrer ID und ihrem Benutzernamen an sowie Datum und Uhrzeit ihrer Erstellung.

```
bin/console mapbender:user:list

User #3 name: max_mustermann since 2019-10-14 12:10:44
```

5.3.4 Datenbanken

bin/console mapbender:database:upgrade

Aktualisiert die Kartenelementkonfigurationen, falls neue vorhanden sind.

```
bin/console mapbender:database:upgrade

Updating map element configs
Found 28 map elements
28/28 [============================] 100%
Updated 28 Map elements
Exiting now
```

bin/console doctrine:database:create

Der Befehl wird einmalig bei der Installation verwendet und legt die Administrationsdatenbank für Mapbender an. Die Angabe zur Datenbankverbindung befindet sich in der parameters.yaml-Datei.

```
bin/console doctrine:database:create
```

bin/console doctrine:schema:create

Mit dem Befehl wird bei der Installation das Datenbankschema angelegt, d.h. es werden die von Mapbender benötigten Tabellen erstellt.

```
bin/console doctrine:schema:create
```

bin/console doctrine:schema:update

Dieser Befehl wird beim Update auf eine Mapbender Version benötigt. Er generiert die SQL-Befehle, um die Datenbank auf das Schema der neuen Version zu aktualisieren.

Note: Die Mapbender-Datenbank sollte eine eigenständige Datenbank sein. Ihre Geodaten sollten in anderen separaten Datenbanken gespeichert werden. Bei einem Update der Mapbender-Datenbank können Tabellen, die nicht durch ein Doctrine Schema beschrieben sind, gelöscht werden. Geodaten in eigenständigen Datenbanken sind davon nicht betroffen.

```
bin/console doctrine:schema:update --complete --dump-sql
bin/console doctrine:schema:update --complete --force
```

- **--dump-sql** Zeigt die Änderungen im Datenbankschema an.
- **--force** Führt die Änderungen im Schema aus.

bin/console doctrine:schema:validate

Der Befehl wird einmalig bei der Installation verwendet. Dieser Befehl überprüft, ob der Aufbau der Datenbank korrekt und aktuell ist.

```
bin/console doctrine:schema:validate
[Mapping]  OK - The mapping files are correct.
```

5.3.5 Druck

bin/console mapbender:print:queue:next

Der Druck in der Warteschlange ist standardmäßig deaktiviert, da er eine externe Integration erfordert. Druckaufträge können danach über die Kommandozeile gesteuert werden. Dafür muss in der parameters.yaml-Datei folgender Parameter hinzugefügt und auf TRUE gesetzt werden:

```
mapbender.print.queueable: true
```

Weitere Informationen zum Warteschleifendruck gibt es im Kapitel *Warteschleifendruck* sowie auf GitHub.

Anschließend wird im *Backend* des Mapbenders der Druckassistent aktualisiert und es erscheinen zwei neue Zeilen, Modus und Warteschleife.

Modus wird auf “Warteschleife” gesetzt und Warteschleife auf “global”, wenn davon auszugehen ist, dass die Druckaufträge für alle Anwender zugänglich sind.

Im Pop-up Fenster des Print Clients erscheint jetzt ein neuer Reiter: “Druckaufträge”. Dieser kann durch folgende Befehle über die Kommandozeile gesteuert werden.

```
bin/console mapbender:print:queue:next
```

Es wird der nächste Druckauftrag ausgeführt, der in der Warteschleife steht. Für einen potenziell unendlich laufenden Prozess können folgende Optionen auf 0 gesetzt werden.

```
bin/console mapbender:print:queue:next --max-jobs=0 --max-time=0
```

Optional kann die Anzahl der Prozesse und die maximale Ausführungszeit limitiert werden.

- **--max-jobs** Limitiert die Prozessanzahl auf einen gewählten Wert.
- **--max-time** Limitiert die maximale Ausführungszeit für den Befehl.

bin/console mapbender:print:queue:rerun

Dieser Befehl führt einen Druckwarteschlangenauftrag erneut aus. Die Angabe der ID ist dabei erforderlich.

```
bin/console mapbender:print:queue:rerun 1

Starting processing of queued job #1
PDF for queued job #1 rendered to /data/mapbender/application/app/../web/prints/mapbender_
↪20191104103745.pdf
```

bin/console mapbender:print:queue:dumpjob

Dieser Befehl gibt Druckaufträge in ein angegebenes Format (JSON oder yml) aus. Die ID des jeweiligen Druckauftrages ist für den Befehl erforderlich. Diese ID kann über die geöffnete Druckwarteschlange in der Mapbender-Anwendung ermittelt werden.

```
bin/console mapbender:print:queue:dumpjob [options] [--] <id>

bin/console mapbender:print:queue:dumpjob 2 > print_configuration.json

bin/console mapbender:print:queue:dumpjob 2

{
        "template": "a4portrait",
        "quality": "288",
        "scale_select": "25000",
        "rotation": "-20",
        "extra": {
                "title": "Egal!"
        },
        "layers": {
                "0": {
                        "type": "wms",
                        "sourceId": "8",
                        "url": "https:\/\/osm-demo.wheregroup.com\/service?_SIGNATURE=31
↪%3AIHZNT0zPZhFG95dN3QOzsizaDwA&TRANSPARENT=TRUE&FORMAT=image%2Fpng&VERSION=1.3.0&EXCEPTIONS=INIMAGE&
↪SERVICE=WMS&REQUEST=GetMap&STYLES=&LAYERS=osm&_OLSALT=0.39407834838362 41&CRS=EPSG%3A25832&BBOX=363375.
↪30907721,5626747.0157598,368124.31589362,5620823.2546257&WIDTH=512&HEIGHT=512",
                        "minResolution": null,
                        "maxResolution": null,
                        "order": 0,
                        "opacity": 1,
                        "changeAxis": false
                },
                "1": {
                        "type": "wms",
                        "sourceId": "7",
                        "url": "https:\/\/wms.wheregroup.com\/cgi-bin\/mapbender_user.xml?_SIGNATURE=26
↪%3Atq6ae-UqhnZLMjiQlLrj-wCHiOI&TRANSPARENT=TRUE&FORMAT=image%2Fpng&VERSION=1.3.0&EXCEPTIONS=INIMAGE&
↪SERVICE=WMS&REQUEST=GetMap&STYLES=&LAYERS=Mapbender_User&_OLSALT=0.6831931928241708&CRS=EPSG%3A25832&
↪BBOX=363375.30907721,5626747.0157598,368124.31589362,5620823.2546257&WIDTH=2400&HEIGHT=1141",
                        "minResolution": null,
                        "maxResolution": null,
                        "order": 0,
                        "opacity": 0.85,
```

(continues on next page)

(continued from previous page)


```
                        "changeAxis": false
                },
                "2": {
                        "type": "wms",
                        "sourceId": "7",
                        "url": "https:\/\/wms.wheregroup.com\/cgi-bin\/mapbender_user.xml?_SIGNATURE=26
↪%3Atq6ae-UqhnZLMjiQlLrj-wCHiOI&TRANSPARENT=TRUE&FORMAT=image%2Fpng&VERSION=1.3.0&EXCEPTIONS=INIMAGE&
↪SERVICE=WMS&REQUEST=GetMap&STYLES=&LAYERS=Mapbender_Names&_OLSALT=0.68319319282417O8&CRS=EPSG%3A25832&
↪BBOX=363375.30907721,5626747.0157598,368124.31589362,5620823.2546257&WIDTH=2400&HEIGHT=1141",
                        "minResolution": null,
                        "maxResolution": null,
                        "order": 1,
                        "opacity": 0.85,
                        "changeAxis": false
                }
        },
        "width": 1920,
        "height": 913,
        "center": {
                "x": 365749.81248542,
                "y": 5623785.1351928
        },
        "extent": {
                "width": 4749.006816409994,
                "height": 5923.761134099215
        },
        "overview": {
                "layers": {
                        "0": "https:\/\/osm-demo.wheregroup.com\/service?_signature=31
↪%3AIHZNT0zPZhFG95dN3QOzsizaDwA&TRANSPARENT=TRUE&FORMAT=image%2Fpng&VERSION=1.3.0&EXCEPTIONS=INIMAGE&
↪SERVICE=WMS&REQUEST=GetMap&STYLES=&LAYERS=osm&CRS=EPSG%3A25832&BBOX=350757.32820012,5616536.5348653,
↪377637.46662208,5629318.6006879&WIDTH=250&HEIGHT=125"
                },
                "center": {
                        "x": 364197.3974111,
                        "y": 5622927.5677766
                },
                "height": 78125,
                "changeAxis": false
        },
        "mapDpi": 90.714,
        "extent_feature": {
                "0": {
                        "x": 362505.8322437394,
                        "y": 5625755.14826519
                },
                "1": {
                        "x": 366968.4389051802,
                        "y": 5627379.404257199
                },
                "2": {
                        "x": 368994.48453732743,
                        "y": 5621812.889632087
                },
                "3": {
                        "x": 364531.877875887,
```


(continues on next page)

(continued from previous page)


```
                        "y": 5620188.63364008
                },
                "4": {
                        "x": 362505.8322437394,
                        "y": 5625755.14826519
                }
        },
        "userId": null,
        "userName": null,
        "legendpage_image": {
                "type": "resource",
                "path": "images\/legendpage_image.png"
        }
}
```


bin/console mapbender:print:runJob

Mit diesem Befehl kann ein Druckauftrag aus einer Druck-Konfigurationsdatei heraus ausgeführt werden. Diese Konfiguration kann über den Befehl bin/console mapbender:print:queue:dumpjob erstellt werden.

```
bin/console mapbender:print:runJob print_configuration.json /tmp/print.pdf
```

bin/console mapbender:print:queue:repair

Wenn ein Druckauftrag in der Warteschlange einen Fehler aufweist oder abgestürzt ist, beispielsweise weil ein WMS-Dienst nicht erreichbar ist, kann der Druck nicht ausgeführt werden.

Mit dem Befehl mapbender:print:queue:repair wird der Status der Druckaufträge zurückgesetzt. Anschließend werden die Aufträge automatisch erneut ausgeführt.

```
bin/console mapbender:print:queue:repair
```

bin/console mapbender:print:queue:clean

Dieser Befehl löscht erfolgreich abgearbeitete Druckaufträge. Dazu zählen einerseits erstellte PDFs als auch dazugehörige Datenbankeinträge zu den Druckaufträgen. Beim Aufruf des Befehls kann die Angabe des Alters hinzugefügt werden, mit der Angabe 20 werden beispielsweise alle Aufträge gelöscht werden, die älter als 20 Tage sind.

```
mapbender:print:queue:clean 20

Print queue clean process started.
Deleted 0 print queue item(s)
```

bin/console mapbender:print:queue:gcfiles

gc steht für “garbage collection”. gcfiles löscht entsprechend alle Druckaufträge, bei denen der Datenbankeintrag keine Referenz mehr zum Dateisystem hat. Dies geschieht zum Beispiel, wenn ein Auftrag in der Datenbank gelöscht oder der Dateipfad zum PDF nicht mehr aktuell ist.

```
bin/console mapbender:print:queue:gcfiles

No unreferenced local files found
```

5.3.6 Mailer

bin/console debug:config framework mailer

Zeigt die/den konfigurierten Mailer an.

```
bin/console debug:config framework mailer
```

5.3.7 Server

symfony server:start –no-tls

Als Voraussetzung für den Symfony-Entwicklungsserver muss zunächst Symfony CLI installiert werden. Im Anschluss führt der Befehl den darin enthaltenen lokalen Webserver aus. Im Terminal wird zusätzlich der Status-Log des Servers mitprotokolliert, während dieser aktiv ist.

Mit Control -C kann der Server wieder gestoppt werden.

Note: Der Symfony Entwicklungsserver kann Einschränkungen unterliegen, die die Performanz der Anwendung in bestimmten Fällen reduziert. Er dient primär Testzwecken.

```
symfony server:start --no-tls
```

```
[OK] Web server listening
     The Web server is using PHP CLI 8.2.10
     http://127.0.0.1:8000
```

Hint: Sie können dem Server die Anwendungsumgebung als `APP_ENV=prod` oder `APP_ENV=dev` mitgeben.

```
APP_ENV=prod symfony server:start --no-tls
```

5.3.8 Cache löschen

bin/console cache:clear

Der Befehl löscht das Cache-Verzeichnis für eine bestimmte Umgebung. Wird keine bestimmte Option angegeben, wird der Cache der dev-Umgebung geleert.

Eventuell muss der Befehl mit root-Rechten (sudo) ausgeführt werden.

Dev-Umgebung:

```
bin/console cache:clear --env=dev
```

Prod-Umgebung:

```
bin/console cache:clear --env=prod --no-debug
```

5.3.9 WMS Dienste

bin/console mapbender:wms:add

Fügt einen neuen WMS in das Mapbender Dienste-Repository hinzu.

```
bin/console mapbender:wms:add https://osm-demo.wheregroup.com/service?VERSION=1.3.0&Service=WMS&
↪request=getCapabilities

* <empty name> OpenStreetMap (WhereGroup)
* * osm OpenStreetMap
* * osm-grey OpenStreetMap (grey scale)
Saved new source #76
```

bin/console mapbender:wms:assign

Fügt einen im Dienste-Repository stehenden WMS-Dienst einer Mapbender-Anwendung hinzu.

```
bin/console mapbender:wms:assign [options] [--] <application> <source> [<layerset>]
```

5.3.10 Konfiguration:

- *application*: ID oder Slug der Anwendung,
- *source*: ID des WMS-Dienstes,
- *layerset* (optional): ID oder Name des Layersets. Der Standardwert ist *main* oder das erste Layerset in der Anwendung.

Folgende zusätzliche Optionen sind möglich:

- **-f, --format** Legt das Format für die GetMap-Anfrage fest, z.B. image/png.
- **-i, --infoformat** Legt das Format für die FeatureInfo-Anfrage fest, z.B. text/html.
- **-p, --proxy** Bestimmt, ob ein Proxy verwendet wird (true | false).
- **-t, --tiled** Gibt an, ob GetMap-Anfragen gekachelt zurückgegeben werden (true | false).

- **-l, --layerorder** Legt die Layer-Reihenfolge fest (standard | reverse) (Hinweis: reverse = QGIS).

bin/console mapbender:wms:parse:url

Befehl zum Parsen des GetCapabilities-Dokuments via URL. Der Befehl kann zum Validieren einer WMS-Adresse verwendet werden.

```
bin/console mapbender:wms:parse:url --validate https://osm-demo.wheregroup.com/service?VERSION=1.3.0&
↪Service=WMS&request=getCapabilities
```

bin/console mapbender:wms:reload:file

Befehl um einen WMS in Mapbender zu aktualisieren. Dabei wird die WMS-ID und eine Datei mit dem getCapabilities-XML angegeben.

```
bin/console mapbender:wms:reload:url 76 /var/www/html/service.xml
```

Folgende zusätzliche Optionen sind möglich:

- **--deactivate-new-layers** Sofern gesetzt, werden neu hinzugekommene Layer in Instanzen, in denen diese vorkommen, deaktiviert. Deaktivierte Layer werden weder in der Karte noch im Ebenenbaum dargestellt.
- **--deselect-new-layers** Sofern gesetzt, werden neu hinzugekommene Layer in Instanzen, in denen diese vorkommen, deselektiert. Nicht ausgewählte Layer werden standardmäßig nicht in der Karte dargestellt, erscheinen aber im Ebenenbaum und können dort vom Benutzer ausgewählt werden.

bin/console mapbender:wms:reload:url

Befehl um einen WMS in Mapbender zu aktualisieren. Dabei wird die WMS-ID und eine Datei mit der getCapabilities-Adresse (URL) angegeben.

```
bin/console mapbender:wms:reload:url 76 https://osm-demo.wheregroup.com/service?VERSION=1.3.0&
↪Service=WMS&request=getCapabilities
```

Folgende zusätzliche Optionen sind möglich:

- **--user=USER** Benutzername (basic auth) [default: ""]
- **--password=PASSWORD** Passwort (basic auth) [default: ""]
- **--deactivate-new-layers** Sofern gesetzt, werden neu hinzugekommene Layer in Instanzen, in denen diese vorkommen, deaktiviert. Deaktivierte Layer werden weder in der Karte noch im Ebenenbaum dargestellt.
- **--deselect-new-layers** Sofern gesetzt, werden neu hinzugekommene Layer in Instanzen, in denen diese vorkommen, deselektiert. Nicht ausgewählte Layer werden standardmäßig nicht in der Karte dargestellt, erscheinen aber im Ebenenbaum und können dort vom Benutzer ausgewählt werden.

bin/console mapbender:wms:show

Befehl zum Anzeigen von Informationen zu einem WMS. Hierbei wird die ID der WMS Datenquelle im Befehl angegeben.

```
bin/console mapbender:wms:show 76

  Source describes 3 layers:
  * <empty name> OpenStreetMap (WhereGroup)
  * * osm OpenStreetMap
  * * osm-grey OpenStreetMap (grey scale)
```

5.3.11 Sonstige

bin/console mapbender:source:rewrite:host

Aktualisiert den Hostnamen in den Quell-URLs, ohne die Funktionen/Capabilities neu laden zu müssen.

```
bin/console mapbender:source:rewrite:host [options] [--] <from> <to>
```

Vergessen Sie nicht, dass Sie sich auch hier weitere Optionen über *Hilfe zu den Befehlen* anzeigen lassen können.

Umsetzungsbeispiel für die Aktualisierung eines Hostnamens:

```
bin/console mapbender:source:rewrite:host "http://osm-demo.wheregroup.com" "https://osm-demo.wheregroup.
↪com"

    3 modified urls in WMS source #5 / OpenStreetMap (OSM) Demo WhereGroup
    Summary:
    1 sources changed
    3 urls changed
    4 sources unchanged
    14 urls unchanged
```

bin/console mapbender:config:check

Der Befehl prüft die Konfiguration und gibt zur Information die Systemkonfiguration aus. Dadurch kann ermittelt werden, ob Abhängigkeiten nicht erfüllt werden.

```
bin/console mapbender:config:check
```

Hint: Bitte beachten Sie, dass der Befehl mapbender:config:check die PHP-CLI Version nutzt. Die Einstellungen der CLI-Version können sich von denen der Webserver PHP-Version unterscheiden. Nutzen Sie beispielsweise php -r 'phpinfo();' zur Ausgabe der PHP-Webserver Einstellungen.

Es werden folgende Anforderungen überprüft und angezeigt:

- Datenbankverbindungen
- PHP-Version
- Systemanforderungen

- Asset-Ordner
- FastCGI
- Apache Modus (rewrite)
- PHP ini
- geladene PHP-Erweiterungen
- Zugriffserlaubnis auf Verzeichnisse

bin/console mapbender:version

Der Befehl gibt die aktuelle Mapbender-Version aus.

```
bin/console mapbender:version

Mapbender 3.0.8.4
```

bin/console debug:config

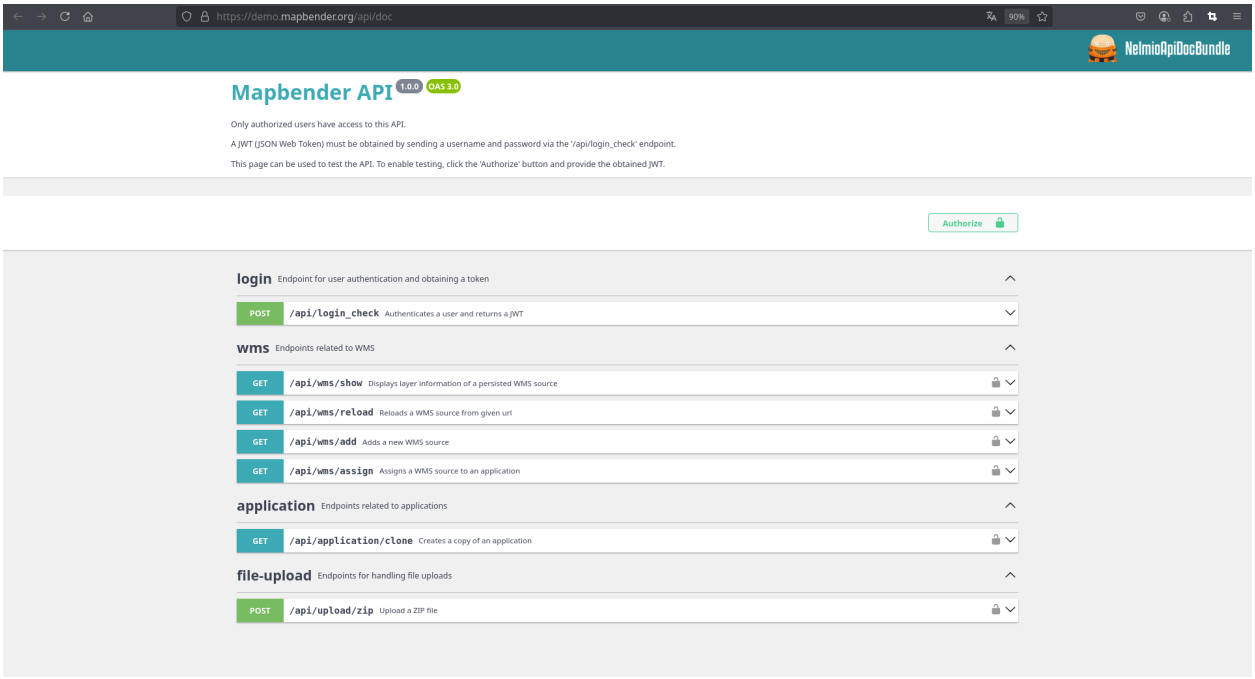
Mit diesem Befehl werden alle registrierten Bundles (Pakete) aufgelistet und, falls vorhanden, der Alias dazu genannt.

```
bin/console debug:config
```

5.4 API

Mapbender bietet eine API, über die Clients verschiedene Befehle ausführen können.

Mit der API kann Mapbender verwaltet werden, ohne die Webadministrationsschnittstelle verwenden zu müssen. Die API bietet Befehle, um beispielsweise Informationen über Dienste zu erhalten, und sie bietet auch Befehle, um Dienste zu veröffentlichen oder zu aktualisieren.



5.4.1 Dokumentation

Die API-Dokumentation ist in jede Mapbender-Installation integriert und über http://localhost/mapbender/api/doc/ öffentlich zugänglich.

In der Dokumentation finden Sie Beispiele für jeden Endpunkt. Bitte beachten Sie, dass Sie sich anmelden und autorisieren müssen, um die Beispiele auszuführen zu können. Außerdem benötigen Sie das Recht „API“ (siehe ACL).

Sie können die Dokumentation in der Mapbender-Demo erreichen unter:

https://demo.mapbender.org/api/doc/

Weitere Informationen zur Developer Dokumentation finden Sie hier:

https://github.com/mapbender/mapbender/blob/develop/docs/api/setup.md

5.4.2 Apache Autorisierung

Apache leitet per default den Authorization Header aus Sicherheitsgründen nicht an den Client weiter. Dies ist für die Nutzung der API aber notwendig. Es muss daher folgendes im VirtualHost oder der Konfiguration gesetzt werden:

```
SetEnvIf Authorization "(.*)" HTTP_AUTHORIZATION=$1
```

5.4.3 Upload Directory ändern

Um das Upload Directory zu ändern, können Sie den

```
api_upload_dir: /data/qgis_server_projects/
```

Parameter in der *paramter.yaml* Datei anpassen.

5.4.4 API-Seite für den öffentlichen Zugriff sperren

Um den öffentlichen Zugriff der API-Seite zu sperren, kann in der Datei *security.yaml* der folgende Parameter angepasset werden.

```
{ path: ^/api/doc, roles: PUBLIC_ACCESS }
```

in

```
{ path: ^/api/doc, roles: ROLE_ADMIN }
```

CHAPTER

SIX

FAQ - HÄUFIG GESTELLTE FRAGEN

6.1 Allgemeines

6.1.1 Umgebungen: prod und dev

F: Wozu sind diese Umgebungen da?

A: Für den produktiven Einsatz rufen Sie Mapbender über die **prod**-Umgebung auf. Erst wenn Sie selbst etwas (an den Twig-, CSS- oder JS-Dateien) entwickeln, nutzen Sie den Aufruf über die **dev**-Umgebung. Der dahinterstehende Entwicklungsmodus gibt mehr Informationen aus, indem er z. B. detailliertere Fehlermeldungen anzeigt.

Mehr Details zu den Umgebungen gibt es im Kapitel *Starten von Mapbender als Produktivumgebung*.

6.1.2 Cache

F: Was ist der Cache und wann muss ich ihn löschen?

A: Der Cache ist ein Zwischenspeicher, aus dem Mapbender auf häufig benutzte Dateien zugreift. Der Cache kann ohne Weiteres gelöscht werden. Sie löschen dabei jedoch nur die Inhalte innerhalb des *mapbender/var/cache/* Ordners. In der Mapbender-Verzeichnisstruktur sind das das *prod*- und - falls vorhanden - das *dev*-Verzeichnis.

Diese zwei Verzeichnisse können ohne Bedenken gelöscht werden. Beim nächsten Aufruf von Mapbender werden im Cache der entsprechenden Umgebung erneut Dateien abgelegt.

Mehr Details zum Cache gibt es im Kapitel *Produktions- und Entwicklungsumgebung und Caches*.

6.1.3 Dienste in Anwendungen

F: In welchen meiner Anwendungen ist ein bestimmter WMS-Dienst eingebaut?

A: Diese Information ist für angemeldete Nutzer mit entsprechenden Berechtigungen über das *Backend* im Bereich *Datenquellen (Sources)* verfügbar:

- Gehen Sie im oberen Menü zu Datenquellen,
- Wählen Sie per Klick auf **Metadaten anzeigen** eine Datenquelle aus und klicken anschließend auf den Reiter **Anwendungen**,
- Hier sehen Sie, in welchen Anwendungen sich der Dienst befindet. Zusätzlich gibt der Reiter über ein entsprechendes Symbol Informationen darüber aus, ob der Dienst als freie oder private Instanz eingebunden wurde und ob die Datenquelle aktiviert oder deaktiviert ist.

- Per Klick auf den Anwendungs- oder Datenquellentitel gelangen Sie bequem zur Konfiguration der Layersets bzw. der Dienst-Instanz.

Wenn Sie SQL bevorzugen, können Sie die WMS-ID der Datenquelle im folgenden SQL unter <id_of_the_wms> eintragen und die Abfrage ausführen, um sich die auf den Dienst zugreifenden Anwendungen tabellarisch anzeigen zu lassen:

```
SELECT mb_core_application.* from mb_core_application, mb_core_layerset, mb_core_sourceinstance, mb_wms_
↪wmsinstance, mb_wms_wmssource, mb_core_source where
-- applications and their layersets
mb_core_application.id = mb_core_layerset.application_id and
-- layersets and their instances
mb_core_layerset.id = mb_core_sourceinstance.layerset and
-- layerset-instances and wms-instance
mb_core_sourceinstance.id = mb_wms_wmsinstance.id and
-- wms-instance and wms-source
mb_wms_wmsinstance.wmssource = mb_wms_wmssource.id and
-- wms-source and mb-core source
mb_wms_wmssource.id = mb_core_source.id and
mb_core_source.id = <id_of_the_wms>;
```

6.1.4 QGIS2Mapbender Plugin

Q: Gibt es eine Möglichkeit, mein QGIS-Projekt direkt aus QGIS als QGIS Server-Projekt und in Mapbender zu veröffentlichen?

A: Ja, die gibt es! Ab der Mapbender Version 4.0 können Sie das Plugin QGIS2Mapbender nutzen. Es überträgt QGIS-Projekte auf ihren Server und stellt sie als WMS bereit. Außerdem können Sie ihr Projekt direkt in Mapbender in eine Anwendung laden. Schauen Sie sich dazu den Blogbeitrag Mapbender-Anwendungen einfach direkt aus QGIS heraus erstellen an. Das Plugin finden Sie auf GitHub.

6.2 Performance

6.2.1 Fehlermeldung beim Kopieren von Anwendungen

F: Ich habe eine komplexe Anwendung, die ich kopieren möchte, was aber fehlschlägt. Woran liegt das?

A: Eine mögliche Ursache kann sein, dass PHP das Arbeiten mit großen Dateien nicht erlaubt. Das tritt vor allem bei FastCGI auf. Dafür dient der Parameter `MaxRequestLen`, den Sie in der *FastCGI*-Konfigurationsdatei anpassen können.

```
# mod_fcgi.conf (Windows)
# set value to 2 MB
MaxRequestLen = 2000000

# fcgid.conf (Linux)
# set value to 2 MB
MaxRequestLen 2000000
```

Analog dazu können Sie die Werte in der *php.ini*-Datei überprüfen:

```
max_execution_time = 240
memory_limit = 1024M
upload_max_filesize = 2M
```

6.2.2 Maximale WMS-Kachelgröße für Druck und Export

F: Wenn ich eine Karte mit einem WMS-Dienst als Bild exportieren oder drucken möchte, erscheint der Dienst anschließend nicht in meiner Datei. Woran kann das liegen?

A: Dies kann verschiedene Gründe haben. Unter bestimmten Umständen kann die angeforderte Pixelausdehnung für den WMS zu groß sein, sodass der Dienst in diesem Fall keine Bilder mehr liefert.

In diesem Fall fügen Sie zu Ihrer *parameters.yaml* folgenden Parameter hinzu, wobei der hier eingetragene Standardwert dem Dienst entsprechend konfiguriert werden kann:

```
mapbender.imaageexport.renderer.wms.max_getmap_size: 8192
```

Durch diesen werden die größtmöglichen `WIDTH=`- und `HEIGHT=`-Werte für WMS-Druck/Export-Anfragen festgelegt. Im GetCapabilities-Request des jeweiligen Dienstes wird die maximale Auflösung unter `MaxWidth` bzw. `MaxHeight` definiert, sodass der getCapabilities-Request das Limit bereits vorgibt. Die Parameter können auch unabhängig voneinander definiert werden:

Weisen Sie zur Veränderung der Breite diesem Parameter ihnen einen Wert zu:

```
mapbender.imaageexport.renderer.wms.max_getmap_size.x:
```

Weisen Sie zur Veränderung der Höhe diesem Parameter einen Wert zu:

```
mapbender.imaageexport.renderer.wms.max_getmap_size.y:
```

6.2.3 Problem bei WMS-Diensten mit vielen Layern

F: Beim Laden eines WMS mit vielen Layern (über 100) in eine Anwendung werden in der layerset-Konfiguration nur Teile der Layer übernommen und angezeigt. Die WMS-Instance kann außerdem nicht abgespeichert werden. Gibt es einen Weg, den WMS dennoch zu verwenden?

A: Mittels des PHP-Parameters max-input_vars kann die Zahl der Eingabe-Variablen erhöht werden. Der Standardwert liegt bei 1000. Die Zahl der Eingabe-Variablen ist bei einem WMS mit vielen Layern sehr hoch, vergleichbar mit der Anzahl der Auswahlmöglichkeiten innerhalb des WMS-Instance-Dialogs. Setzen Sie bei der Arbeit mit großen WMS mit vielen Layern den Parameter hoch, beispielsweise auf **2000**. Die Zahl hängt direkt mit der Anzahl der Layer im WMS zusammen.

```
;; 1000 (default)
max_input_vars = 2000
```

6.3 Installation

6.3.1 Fehlermeldung beim Drucken

F: Ich bekomme einen Fehler beim Drucken. Das Log (*var/log/prod.log*) wirft folgende Fehlermeldung:

```
CRITICAL - Uncaught PHP Exception Symfony\Component\Debug\Exception\UndefinedFunctionException:
"Attempted to call function "imagecreatefrompng"
from namespace "Mapbender\PrintBundle\Component"."
at /srv/mapbender-starter/application/mapbender/src/Mapbender/PrintBundle/Component/PrintService.php
↪line 310
```

A: Bitte stellen Sie sicher, dass Sie die php-gd-Bibliothek installiert haben. Wir empfehlen, die Erweiterung bereits vor dem Download von Mapbender zu installieren. Auf Linux-Systemen können Sie sie wie folgt nachinstallieren:

```
sudo apt-get install php-gd
```

6.3.2 SSL-Zertifikatsfehler

F: Ich bekomme einen SSL-Zertifikatsfehler angezeigt, wie kann ich diesen beheben?

A: Das Problem kann beim Zugriff auf einen Dienst über https auftreten. Wenn beim Laden oder Aktualisieren einer OGC WMS-Datenquelle auf Windows-basierten Mapbender-Servern ein SSL-Zertifikatsproblem auftritt, müssen Sie die *cacert.pem*-Datei aktualisieren und in Ihrer *php.ini* auf sie verweisen.

Der Fehler sieht wie folgt aus:

```
cURL error 60: SSL certificate problem: unable to get local issuer certificate
```

Note: Es gibt eine *cacert.pem*-Datei, die alle vertrauenswürdigen Zertifizierungsstellen auflistet. *cacert.pem* ist base64-kodiert und definiert alle vertrauenswürdigen Zertifizierungsstellen. Sie können die Datei hier herunterladen.

Der Fehler tritt auf, wenn die Datei nicht in der *php.ini* referenziert oder die *cacert.pem* nicht aktuell ist.

Verweisen Sie auf *cacert.pem* in der *php.ini*, um das Problem zu beheben:

```
curl.cainfo="C:\[Ihr Pfad]\cacert.pem"

openssl.cafile="C:\[Ihr Pfad]\cacert.pem"
```

Wenn Sie ein individuelles, selbstsigniertes Zertifikat verwenden, können Sie die Informationen Ihrer Zertifizierungsstelle in der Datei *cacert.pem* hinzufügen.

Weitere Informationen finden Sie in der PHP-Dokumentation.

6.3.3 Symfony Abhängigkeiten nachinstallieren

F: Nach dem Update von Mapbender muss für meine Installation eine Symfony Komponente nachinstalliert werden. Wie erreiche ich das?

A: Es ist möglich, Symfony Komponenten über die Kommandozeile manuell nachzuinstallieren. Dies geschieht mithilfe des Befehls

```
./bin/composer install symfony/your-bundle
```

Ersetzen Sie einfach `your-bundle` mit dem Komponentennamen.

Im GitHub-Symfony-Projekt finden Sie eine entsprechende Auflistung über die Abhängigkeiten.

6.3.4 Warnungen im composer oder bootstrap Skript

F: Ich bekomme beim Ausführen von `bootstrap` bzw. von `composer update` eine Deprecation-Warnung:

```
Deprecation Notice: The callback ComposerBootstrap::checkConfiguration declared at
/srv//mapbender-starter/application/src/ComposerBootstrap.php accepts a Composer\Script\CommandEvent
but post-update-cmd events use a Composer\Script\Event instance.
Please adjust your type hint accordingly, see https://getcomposer.org/doc/articles/scripts.md#event-
↪classes
in phar:///srv/mapbender-starter/composer.phar/src/Composer/EventDispatcher/EventDispatcher.php:290
```

A: Das ist abhängig von der PHP-Version, auf der Sie diese Kommandos aufrufen und taucht bei PHP Versionen < 7 auf. Je nach Mapbender-Release empfehlen wir unterschiedliche PHP-Versionen, die die Warnungen nicht auslösen.

6.4 Entwicklung

6.4.1 Updates von Modulen

F: Wie kann ich einen speziellen Branch des Mapbender-Moduls auschecken und testen? Wie bekomme ich das wieder rückgängig? Hilft mir Composer dabei?

A: Möglichkeit 1 (über Git): Über die Konsole in das Verzeichnis *application/mapbender* gehen und den gewünschten Branch auschecken. Nach dem Testen wieder den aktuellen Branch auschecken. Leeren Sie zwischendurch das Symfony-Cache-Verzeichnis.

Möglichkeit 2 (über Composer): "mapbender/mapbender": "dev-fix/meinfix" eintragen und ein Composer Update ausführen. Dabei werden aber auch alle anderen Vendor-Pakete aktualisiert. Rückgängig kann dies mit der Angabe des vorherigen Branches gemacht werden: Dazu erneut in *application/mapbender* gehen und den Branch auschecken.

6.4.2 Überschreiben von Twig-Dateien

F: Wie kann ich Twig-Dateien in Bundles überschreiben und auf diese Weise das Design bestimmter Elemente anpassen?

A: Um Twig-Dateien zu überschreiben, legen Sie einfach eine Twig-Datei mit dem gleichen Namen im Verzeichnis *templates/bundles/<bundlename>* ab. Wenn Sie z.B. das Erscheinungsbild der **Koordinatenanzeige** anpassen möchten (zu finden unter *Resources/views/Element/coordinatesdisplay.html.twig* im Mapbender CoreBundle), erstellen Sie eine Kopie, passen diese an und legen sie unter *templates/bundles/MapbenderCoreBundle/Element/coordinatesdisplay.html.twig* ab. Diese neue Datei wird anstelle der ursprünglichen verwendet.

6.5 Oracle

6.5.1 Einstellungen für die Oracle Datenbank - Punkt und Komma

F: Ich bekomme bei Oracle einen Fehler, wenn ich `doctrine:schema:create` ausführe. Warum? Hier ist meine Fehlermeldung:

```
[Doctrine\DBAL\Exception\DriverException]
An exception occurred while executing
'CREATE TABLE mb_wms_wmsinstance (id NUMBER(10) NOT NULL,
                                  [...]
                                  PRIMARY KEY(id))':
ORA-01722: Ungültige Zahl
```

A: Wahrscheinlich kommt Oracle nicht mit den Dezimaltrennern zurecht und erwartet ein Komma statt einem Punkt (also 1,25 statt 1.25). Das Einsetzen des nachfolgenden Statements am Ende der `doctrine.yaml` verhindert dies (Cache danach leeren):

```
services:
  oracle.session.listener:
    class: Doctrine\DBAL\Event\Listeners\OracleSessionInit
    tags:
      - { name: doctrine.event_listener, event: postConnect }
```

Es handelt sich dabei um die Verknüpfung zu einer Service-Klasse, die von Doctrine bereitgestellt wird. Die setzt nach der Verbindung zur Datenbank Session-Variablen (ALTER SESSION), sodass PHP und Oracle zusammenarbeiten können.

Ursachen können sein: Länder- und Spracheinstellungen des Betriebssystems (z. B. unter Windows), Einstellungen des Oracle-Clients, Einstellungen während der Installation von Oracle.

6.5.2 Rechtevergabe bei der Oracle-Datenbank

F: Welche Rechte benötigt der Mapbender User auf der Oracle-Datenbank?

A: Mapbender benötigt Zugriff auf:

```
- Create Sequence
- Create Session
- Create Table
- Create Trigger
- Create View
```

6.5.3 Langsamer Zugriff auf Oracle-Datenbanken

F: Beim Zugriff auf Oracle-Datenbanken reagiert Mapbender teilweise recht langsam, Abfragen dauern länger als gewöhnlich. Was kann ich anpassen?

A: Es gibt zwei Parameter in der *php.ini*, mit der die Zugriffe auf die Oracle Datenbanken verbessert werden können: oci8.max_persistent und oci8.default_prefetch. Passen Sie diese an.

```
oci8.max_persistent = 15
oci8.default_prefetch = 100000
```

Des Weiteren stellen Sie in der *doctrine.yaml* in der jeweiligen Datenbank-Verbindung den persistent-Parameter auf `true`.

```
persistent=true
```

CHAPTER

SEVEN

ARCHITEKTUR

7.1 Verzeichnisstruktur in Mapbender

7.1.1 application

Bei einer *Git-basierten Installation* enthält das *application*-Verzeichnis beinahe sämtliche Mapbender-Komponenten. Wichtige Unterverzeichnisse und Dateien sind:

- Das *config*-Verzeichnis,
- Der Anwendungskernel (*src/Kernel.php*) als Zugangspunkt zu Symfony,
- Autoloading (*vendor/autoload.php*) generiert von Composer,
- Kommandozeilen-Anwendungen für Pflege und Management (`bin/console`, `bin/phpunit`) und
- *.env* und *.env.local*: Globale Konfigurationsdateien, z.B. für Datenbank-URLs und SMTP-Server-Verbindungen.

bin

Hier liegen diverse Bibliotheken.

config

Grundlegende Konfigurationsdateien von Mapbender liegen im Verzeichnis *config/* sowie im Verzeichnis *config/packages*. Diese Dateien sind dabei von besonderer Bedeutung:

- *doctrine.yaml*
- *parameters.yaml*

config/applications

Als YAML definierte Anwendungen können in dem Verzeichnis *config/applications* abgelegt werden. Die bekannten Beispielanwendungen **Mapbender mobile**, **Mapbender Demo Map** und **Mapbender Demo Map basic** liegen dort als einzelne YAML Dateien.

Weitere Informationen im Kapitel *YAML Konfiguration (Konfigurations- und Anwendungsdateien)*.

mapbender

- Verzeichnis des Mapbender Submoduls. Liefert die Mapbender-spezifischen Bundles und den Mapbender-Code.
- Verzeichnis für spezielle Anwendungsressourcen (*Resources/*)

mapbender/…../translations

Die Übersetzungen werden in YAML-Dateien gespeichert. Jede Sprache benötigt eine YAML-Datei, wie z.B. *messages.fr.yaml* für die französische Übersetzung.

- Verzeichnis: *mapbender/src/Mapbender/CoreBundle/Resources/translations/*

public

Dieses Verzeichnis muss vom Webserver veröffentlicht werden. Ein entsprechendes Alias muss auf dieses Verzeichnis verweisen.

Es kontrolliert:

- *index.php*: FrontendController (PHP-Script, welches aufgerufen werden kann). Benutzt je nach Definition der Variable `APP_ENV` den Produktiv- oder Entwicklungsmodus (siehe *.env bzw. .env.local*).
- *index_dev.php*: Benutzt immer die Entwicklungsumgebung. Kann standardmäßig nur von der lokalen Hostumgebung aufgerufen werden. Lesen Sie mehr dazu, wie auch andere IPs für den Zugriff freigegeben werden können, unter *Produktions- und Entwicklungsumgebung und Caches*.
- *public/* beinhaltet öffentlich verfügbare Ressourcen wie css, js, favicon etc.

public/bundles

- Hier sind die statischen Ressourcen der einzelnen Bundles gespeichert.
- Das folgende Kommando kopiert die Ressourcen der Bundles in den Ordner:

```
bin/console assets:install --symlink --relative public
```

Note: **Hinweis**: Wenn Sie Windows benutzen, können Sie keine symbolischen Links verwenden. Daher müssen Sie das Kommando `php.exe bin/console assets:install public` nach jeder Änderung von JavaScript-, css- oder twig- sowie Bilddateien aufrufen, um die Dateien in das öffentliche Verzeichnis zu kopieren.

src

Verzeichnis für anwendungsspezifische Bundles (Kunden-Bundle).

- Der Anwendungs-Kernel (*src/Kernel.php*) wird über die FrontendController aufgerufen; darüber wird die gesamte Anwendung kontrolliert.

var

Dieses Verzeichnis beinhaltet:

- die php-Caches (*var/cache/dev* and *var/cache/prod*),
- das log-Verzeichnis (*var/log*),
- das SQLite-Datenbank-Verzeichnis (*var/db/*).

vendor

Verzeichnis mit externen Bibliotheken, die via Composer geladen wurden. Ressourcen werden von Symfony durch das Autoladen verwendet:

- Autoladen-Datei (*autoload.php*)

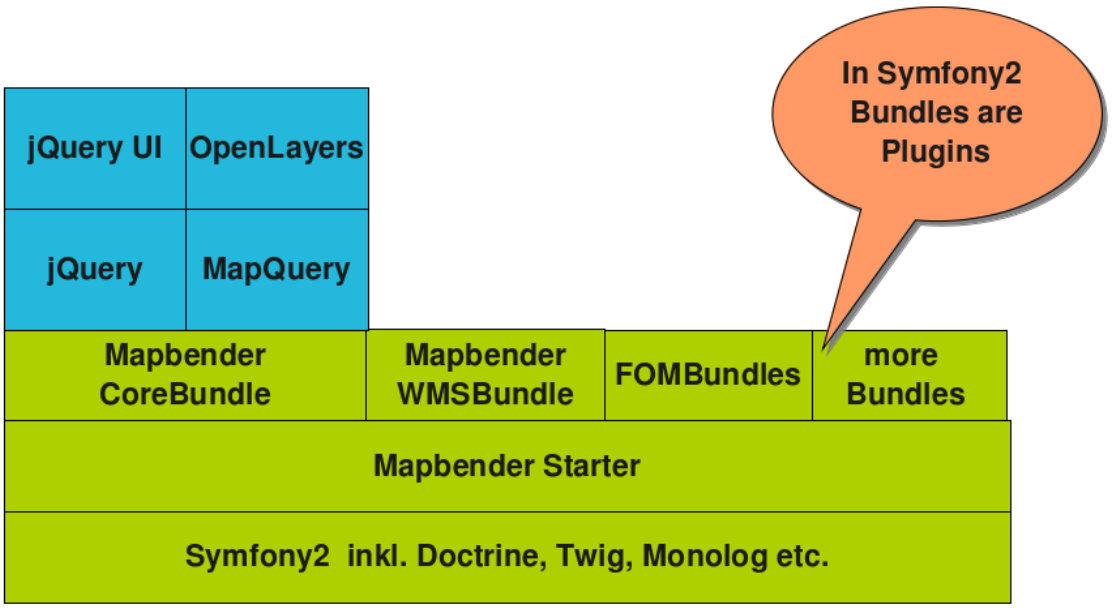
7.2 Komponenten

Mapbender besteht aus unterschiedlichen Software-Komponenten. Serverseitig wird Symfony als Framework mit seinen integrierten Bestandteilen (Doctrine, Twig, Monolog) genutzt.

Clientseitig nutzt Mapbender OpenLayers, MapQuery, jQuery und jQuery UI.

Wir stellen das Mapbender Core Bundle mit den Mapbender-Funktionalitäten bereit. Darüber hinaus gibt es weitere optionale Bundles.

Außerdem bieten wir das Mapbender Starter Paket an. Mit diesem ist Mapbender leicht installierbar.



7.2.1 Symfony

Symfony ist ein objektorientiertes PHP Web Development Framework. Es gilt als Standard für den Bau von Webseiten und Web-Applikationen.

Diese Komponenten bietet Symfony an:

- Symfony config.php zum Prüfen von Voraussetzungen
- Symfony Profiler
- Datenbankabstraktion mit Doctrine
- Benutzerauthentifizierung & -autorisierung
- Templating mit Twig
- Übersetzungen über xliff-files
- Logging mit Monolog
- Sicherheitskomponenten

Mehr über Symfony und dessen Funktionsweisen findet sich in der offiziellen Dokumentation:

- https://symfony.com/doc/current/index.html

7.2.2 OpenLayers

OpenLayers ist eine JavaScript-Bibliothek, die das Anzeigen von Geodaten im Webbrowser ermöglicht.

Mehr über OpenLayers findet sich auf der offiziellen Website unter: https://openlayers.org/

7.2.3 jQuery and jQuery UI

jQuery ist eine Standard-JavaScript-Bibliothek. jQuery UI bietet eine Sammlung von Benutzeroberflächenelementen auf Grundlage der jQuery JavaScript Bibliothek an.

Mehr über jquery: https://jquery.com

Mehr über jquery UI: https://jqueryui.com/

7.2.4 Mapbender

Mapbender ist aus Komponentensicht eine Bundle-Sammlung. Als Voraussetzung müssen das MapbenderCoreBundle und das FOMBundle genutzt werden.

Andere (optionale) Bundles, die die Funktionalität grundlegend erweitern, sind:

- WMSBundle
- WMTSBundle
- WMCBundle
- MonitoringBundle

CoreBundle

Das Mapbender CoreBundle ist das Mapbender Standard-Bundle. Es beinhaltet grundlegende Klassen für Anwendungen, Elemente, Layer und mehr.

Es stellt jQuery, jQuery UI, OpenLayers und MapQuery für alle anderen Mapbender Bundles bereit.

7.2.5 Mapbender Starter

Mapbender Starter ist ein Symfony Demo-Projekt welches Mapbender Bundles nutzt, um eine grundlegende Mapbender-Anwendung zu präsentieren.

Es enthält die Standard-Demo-Anwendungen, welche in der mapbender.yaml definiert werden. Diese sind über ein Web-Interface mit Authentifizierungsmöglichkeiten eingebunden. Das *Backend* kann darüber hinaus genutzt werden, um Anwendungen, Benutzer und Gruppen zur Rechteverwaltung zu erstellen und verschiedene Dienste einzubinden.

Mapbender Starter kann als einfache Startmöglichkeit in das eigene Mapbender-Projekt genutzt werden.

Link zum GitHub-Repository: https://github.com/mapbender/mapbender-starter

7.2.6 Externe Repositories

Noch mehr mit Mapbender zusammenhängende Repositories finden sich auf GitHub. Verschiedene Anbieter können auf dieser Plattform Mapbender-spezifische Bundles anbieten; z.B. das DesktopIntegrationBundle, welches von der WhereGroup bereitgestellt und von Kunden gesponsert wird.

Die WhereGroup bietet Mapbender-Bundles in GitHub an: https://github.com/WhereGroup

7.3 Konzept

Dieses Kapitel gibt eine Einführung in das Konzept von Mapbender. Nach dem Lesen sollten Sie den Aufbau von Mapbender und die Interaktion zwischen den verschiedenen Komponenten der Software verstehen.

7.3.1 Anwendung

Eine Anwendung (Applikation) ist eine einzelne Mapbender-Konfiguration, die normalerweise als interaktive Webkarte verwendet wird.

Jede Anwendung besteht aus verschiedenen Komponenten:

- Elemente
- Layersets
- Anwendungsvorlagen

7.3.2 Elemente

Elemente sind die Bausteine der Anwendung. Jedes Element liefert Funktionalität und kann ggf. mit anderen Elementen interagieren. Das Kartenelement ist vermutlich das Element, das Sie am meisten benutzen werden.

Jedes Element besteht aus vier Bereichen:

- PHP-Klasse: Sie beschreibt das Element mit seinen Möglichkeiten. Die Klasse kann auch ein Ajax callback point bereitstellen, so dass das clientseitige Widget (siehe unten) Datenbankabfragen ausführen kann und die Ergebnisse beispielsweise anzeigen kann.
- JavaScript-Widget: Dies ist der clientseitige Teil des Elements. Dieser Teil beinhaltet alles, was Sie am Bildschirm ausführen und mit dem Sie interagieren. Über Ajax kann dieser Teil den serverseitigen Gegenpart aufrufen und beispielsweise Datenbankabfragen ausführen.
- Template: HTML, welche das Element verwendet. In der einfachsten Version würde dies nur ein DIV-Tag sein. Aber es kann auch komplexer sein.
- CSS: Dient dem Styling der Elemente.

7.3.3 Anwendungsvorlage

Jede Anwendung ist eine HTML-Seite. Die Anwendungsvorlage definiert das Basislayout der Seite. Die Anwendungen können unterschiedliche Vorlagen verwenden, beispielsweise spezielle Vorlagen für die mobile Anwendung.

7.3.4 Frontend

Das Frontend ist die Anwendungssicht der Mapbender-Installation. Es beinhaltet die Anwendungsliste und jede Sicht auf die Anwendung. Der Anwender interagiert mit dem Frontend.

7.3.5 Backend

Über das *Backend* wird die Mapbender-Installation konfiguriert. Es wird auch der Manager genannt. Mit ihm lassen sich Benutzer, Gruppen und Rollen, Dienste, Anwendungen und Elemente verwalten.

CHAPTER

EIGHT

ENTWICKLUNG

8.1 Einführung

Diese Dokumentation ist an Mapbender-Entwickler gerichtet und behandelt Themen, die von Mapbender-Administratoren und -Benutzern nicht benötigt werden.

8.1.1 Was ist zu beachten?

Warning: Dieses Dokumentationskapitel ist teilweise stark veraltet. Eine aktuelle technische Dokumentation (englischsprachig) ist direkt in den Mapbender-Quellcode integriert. Sie kann direkt auf GitHub gelesen werden.

Sie sollten einige Dinge wissen, um an der Mapbender-Entwicklung mitwirken zu können:

- Objektorientiertes PHP: Verwendet wird PHP, welches objektorientierte Programmierung ermöglicht.
- Symfony: Mapbender baut auf Symfony auf. Folgen Sie der Symfony-Dokumentation, dort erfahren Sie mehr.
- JavaScript: jQuery wird häufig verwendet, vor allem kommt die jQuery UI Widget Factory häufig zum Einsatz. Das Verständnis davon ist essentiell, um JavaScript-Code zu schreiben.

8.1.2 Installation

Die Installation aus den Git-Quellen heraus wird unter *Git-basierte Installation* beschrieben.

8.1.3 Module und Bundles

Bitte beachten Sie die jeweiligen Beiträge im Entwicklungshandbuch CONTRIBUTING.md, um zu erfahren, was ein Modul und was ein Bundle ist und wie letzteres erstellt werden kann.

8.1.4 Twig

Mapbender nutzt den Template Ansatz, der von Symfony bereitgestellt wird. Über diesen kann Symfony beispielsweise HTML oder CSS erzeugen. Ein Template ist eine Textdatei, die jedes textbasierte Format wie HTML oder XML generieren kann. Sie kann verwendet werden, um ein Layout zu erstellen. Auf diese Weise kann ein Basislayout erstellt und dann beliebige Layout-Blöcke mit individuellen Templates überschrieben oder hinzugefügt werden. Lesen Sie mehr über Templates auf der Seite *Wie werden eigene Style-Vorlagen (templates) erzeugt?* oder im Contributing Guide. Eine Einführung in Twig gibt außerdem die Symfony Template Dokumentation.

8.1.5 Überschreiben von JavaScript- und CSS/Sass-Ressourcen

Unter Verwendung der ApplicationAssetService-Klasse sind JavaScript- und CSS/Sass-Ressourcen manuell überschreibbar:

Nutzen Sie dazu `mapbender.application_asset.service` innerhalb einer Klasse, zum Beispiel in der *boot*-Methode Ihrer Bundle-Datei mit `$this->container->get('mapbender.application_asset.service')`. Alternativ können Sie dieses Verhalten in einer beliebigen PHP-Datei mit `<argument type="service" id="mapbender.application_asset.service" />` erreichen. Stellen Sie bei dieser Methode wiederum sicher, dass Sie eine Datei verwenden, die aufgerufen wird (z.B. das Template).

Rufen Sie anschließend `ApplicationAssetService::registerAssetOverride` oder `ApplicationAssetService::registerAssetOverrides` auf, um Assets für den Ersatz zu markieren. Nachfolgend finden Sie ein Beispiel, welches eigene Ressourcen für die **Button**-Klasse nutzt:

```
class MyBundle extends Bundle
{
   [ ... ]

   public function boot(): void
   {
      parent::boot();
      $assetService = $this->container->get('mapbender.application_asset.service');
      $assetService->registerAssetOverride('@MapbenderCoreBundle/Resources/public/sass/element/button.
↪scss', '@MyBundle/Resources/public/element/my_button.scss');

      $assetService->registerAssetOverrides([
            '@MapbenderCoreBundle/Resources/public/sass/element/button.scss' => '@MyBundle/Resources/
↪public/sass/element/my_button.scss',
            '@MapbenderCoreBundle/Resources/public/js/element/button.js' => '@MyBundle/Resources/public/
↪js/element/my_button.js',
      ]);
   }
}
```

Hint: Alternativ ist es möglich, Ressourcen mithilfe *eines Parameters* zu überschreiben.

8.1.6 Wo gibt es Hilfe?

Technische Dokumentationen

- Mapbender

Mailinglisten

- Mapbender-Developer und -Anwender Mailingliste

Bibliotheken und Frameworks

- Symfony framework
- Composer documentation
- General GitHub documentation
- GitHub pull request documentation

8.1.7 Themen

Elemente verstehen

> **Warning:** Dieses Dokumentationskapitel ist teilweise stark veraltet. Eine aktuelle technische Dokumentation (englischsprachig) ist direkt in den Mapbender-Quellcode integriert. Sie kann direkt auf GitHub gelesen werden.

Bereiche der Elemente

PHP Class

TODO

Twig Template

Jedes Element benötigt ein HTML-Element. In den meisten Fällen kann das ein DIV-Element sein, aber es kann auch komplexer sein.

Für Mapbender wird Twig verwendet. Eine einfache Twig-Vorlage für ein Element kann wie folgt aussehen:

```
<div id="{{ id }}" class="mb-element mb-element-myclass"></div>
```

Mehrere Angaben müssen gesetzt werden:

- id (wird von Mapbender generiert)
- allgemeine mb-element-Klasse
- spezielle Klasse für das Element

JavaScript Widgets

Element Widgets werden unter Verwendung der jQuery UI widget factory erzeugt. Dies gewährleistet eine einheitliche Struktur für die Widget-Entwicklung und bietet:

- voreingestellte Optionen.
- Konstruktoren und Dekonstruktoren (optional)
- private und öffentliche Methoden.

Das grundlegende Gerüst sieht folgendermaßen aus:

```
(function($) {

// Das ist die Widget Factory. Es wird sowohl die Widget-Klasse "mbMyClass" im jQuery-Objekt als auch eine
//  "mbMyClass" Object im "mapbender"-Namensraum im jQuery-Object  erzeugt (sie werden beide unterschiedlich verwendet). Verwenden Sie ein
// "mb"-Präfix für Ihre Widget-Namen, damit existierende jQuery-Funktionen nicht überschrieben werden.
$.widget('mapbender.mbMyClass', {
            // Es werden voreingestellte Optionen angelegt, die in der Mapbender-Konfiguration überschrieben werden kann.
    //  Es wird später in die PHP-Klasse verschoben.
    // Auf das endgültige options-Objekt kann zugegriffen werden als "this.options".

    options: {
        foo:    'bar',
        answer: 42
    },

    // Dieses Attribut ist privat für Ihr Widget.
    var1: null,

    // Der Konstruktor wird bei der Widget-Initialisierung aufgerufen.
    _create: function() {
        // Hier wird alles für das Setup angelegt, beispielsweise für das Event Handling
        this.element.bind('mbmyclassmagicdone', $.proxy(this._onMagicDone, this));
        this.element.bind('click', $.proxy(this._clickCallback, this));
    },

    // Der Destruktor, wird in diesem Beispiel über jQuery geliefert.
    destroy: $.noop,

    // Öffentliche Funktion, sind beispielsweise über "$('#element-13').mbMyClass('methodA', parameterA, parameterB)" abrufbar
    methodA: function(parameterA, parameterB) {
        this._methodB(parameterA);
    },

    //  Private Funktion, sind nur innerhalb des Widget abrufbar
    _methodB: function(parameterA) {
        // Das ausgelöste Signal wird genannt "mbmyclassmagicdone" (kleingeschrieben)
        this._trigger('magicdone');
    },

    _onMagicDone: function() {
        alert("Oh, magic!");
    },
```

(continues on next page)

(continued from previous page)

```
    _clickCallback: function(event) {
        var target = $(event.target);
        var id = target.attr('id');
        // ...
    }

});

})(jQuery);
```

Für das Event Handling wird jQuery.proxy verwendet, um sicherzustellen, dass ein der Callback im richtigen Kontext gewährleistet wird:

```
// ...

this.element.click($.proxy(this._clickCallback, this));

// ...
```

In diesem Fall ist "this" innerhalb der clickCallback Methode das This, das als der zweite Parameter übergeben wird (in der Regel die Widget Instanz) und nicht das HTML-Element, das das Event angestoßen hat.

Kommunikation zwischen Elementen

Es gibt eine aktive und passive Kommunikation zwischen den Widgets. Die aktive Kommunikation, wird genutzt, um eine öffentliche Methode eines anderen Widget abzurufen. Dazu selektieren Sie das HTML-Element des Widgets mit jQuery und rufen die Methode folgendermaßen auf:

```
var otherElement = $('#element-13').mbMyClass('methodA', parameterA, parameterB);
```

Dies ist eine Standard-jQuery UI Syntax und selbsterklärend. Es ist die Frage, wie Sie die anderen HTML-Elemente erkennen? Um ein Element zu selektieren, wird bevorzugt die ID verwendet. Diese ID's werden jedoch zur Laufzeit von Mapbender generiert, wenn die Anwendung startet, so dass Sie nicht davon ausgehen können, dass die ID immer gleich ist. Glücklicherweise können Sie in der Konfiguration eine Element-ID als eine Target-Option für ein anderes Element übergeben. Diese wird mit der Laufzeit-ID des Target-Elements des HTML-Element überschrieben, so dass Sie in Ihrem Widget-Code auf die richtige ID "this.options.target" zugreifen können.

```
$('#' + this.options.target).mbMyClass('methodA', parameterA, parameterB);
```

Die passive Kommunikation wird verwendet, um Ereignisse anderer Targets anzumelden. Sie müssen das HTML-Element kennen und können nun dem anderen Widget lauschen, um ihr Widget abzurufen. Dieses wird mit Standard-jQuery-Events vorgenommen:

Wenn Sie die "_trigger"-Methode mit jQuery UI Widget Factory bereitstellen …

Wie können eigene Elemente erzeugt werden?

Warning: Dieses Dokumentationskapitel ist teilweise stark veraltet. Eine aktuelle technische Dokumentation (englischsprachig) ist direkt in den Mapbender-Quellcode integriert. Sie kann direkt auf GitHub gelesen werden.

Mapbender bietet einen bin/console-Befehl zur Erzeugung von Elementen.

Hierbei können vier verschiedene Typen von Elementen generiert werden:

- einfache Elemente (general)
- Schaltflächen (buttons)
- Elemente, die auf einen Kartenklick (map-click) reagieren
- Elemente, die auf das Aufziehen eines Rechtecks (map-box) reagieren

Achtung: Die generierten Elemente enthalten lediglich ein Basisgerüst und müssen anschließend noch angepasst werden.

Im Folgenden soll am Beispiel eines map-klick-Elementes das Erzeugen und die Anpassung eines Elementes gezeigt werden.

Die Arbeitsschritte zum Erzeugen von eigenen Elementen

Die Arbeitsschritte auf dem Weg zum eigenen Element.

- Erzeugen Sie ein eigenes Bundle
- Erzeugen Sie ein Element mit Hilfe von bin/console
- Passen Sie das Element an Ihre Bedürfnisse an
- Fügen Sie das Element in die Funktion *getElements()* ein, um es über das *Backend* verfügbar zu machen

Anlegen eines eigenen Bundles mit bin/console generate:bundle

Die Hilfe zum Befehl erhalten Sie über die Option help:

```
bin/console generate:bundle --help
```

```
bin/console generate:bundle --namespace=Workshop/DemoBundle --dir=src
```

Für die Erstellung müssen noch einige Angaben gemacht werden:

```
Bundle name [WorkshopDemoBundle]: WorkshopDemoBundle

Determine the format to use for the generated configuration.
Configuration format (yml, xml, php, or annotation): annotation

To help you get started faster, the command can generate some
code snippets for you.
```

(continues on next page)

(continued from previous page)

```
Do you want to generate the whole directory structure [no]? yes

Summary before generation
You are going to generate a "Workshop\DemoBundle\WorkshopDemoBundle" bundle
in "src/" using the "annotation" format.

Do you confirm generation [yes]? yes

Confirm automatic update of your Kernel [yes]? yes

Confirm automatic update of the Routing [yes]? yes
```

Nach diesen Schritten liegt das neue Bundle im Verzeichnis src vor. Außerdem wurde das Bundle in der Datei Kernel.php registriert. In der Datei routing.yaml wurde eine neue Route für das Bundle eingetragen.

Anpassung des eigenen Elements

Anpassung des Titels und der Beschreibung

In der PHP-Datei finden Sie zahlreiche Funktionen. Ändern Sie den return-Wert der Funktionen *getClassTitle()* und *getClassDescription()*.

```
public static function getClassTitle() {
    return "MapKlick";
}
```

```
public static function getClassDescription() {
    return "Generates an Url with the the mapklick coordinates added";
}
```

Registrierung des neuen Elements

Ein Element kann registriert werden, indem es in der Funktion *getElements()* in der Datei src/Workshop/DemoBundle/WorkshopDemoBundle.php aufgeführt wird. Nach der Erstellung liegt diese Funktion vorerst nicht vor. Fügen Sie diese ein. Außerdem muss die Referenz zum Mapbernder-CoreBundle eingetragen werden (use Mapbender\CoreBundle…). Sie müssen weiterhin angeben, dass die Klasse das MapbenderBundle erweitert.

Durch diesen Eintrag kann das Element im *Backend* bei der Anwendungskonfiguration ausgewählt werden.

```
<?php

namespace Workshop\DemoBundle;

use Symfony\Component\HttpKernel\Bundle\Bundle;
use Mapbender\CoreBundle\Component\MapbenderBundle;

class WorkshopDemoBundle extends MapbenderBundle
{
    public function getElements()
    {
```

(continues on next page)

(continued from previous page)

```
        return array(
            'Workshop\DemoBundle\Element\MapKlick'
        );
    }
}
```

Element zu einer Anwendung hinzufügen

Erstellen Sie eine Anwendung und fügen Sie das neue Element zu der Anwendung hinzu. Sie finden das Element unter dem Element-Titel in der Liste der Elemente. Beachten Sie, dass die anschließende Konfiguration des Elementes im YAML-Syntax erfolgt. Wenn Sie das Karten-Element (map) als *target* verwenden möchten, müssen Sie die ID des Kartenelements ermitteln. Dies kann beispielsweise über Firebug erfolgen.

Ändern der Aktion des Klick-Ereignisses

Wenn Sie ein map-click-Element erzeugen, reagiert dieses auf das Klick-Ereignis mit einer Aktion. Diese Aktion kann modifiziert werden. Schauen Sie sich dazu die JQuery widget Datei an (mapbender/src/Workshop/DemoBundle/public/mapbender.element.mapklick.js).

Hier finden Sie die Funktion *_mapClickHandler()*, die die Koordinaten des Klick-Ereignisses ermittelt und an die Funktion *_mapClickWorker()* weitergibt. Standarmäßig gibt das neu generierte Element die Pixelposition und Koordinate des Klicks in einem Dialog aus.

Sie können die Aktion der Funktion *_mapClickWorker()* anpassen.

Standarddefinition der Funktion mapClickWorker

```
_mapClickWorker: function(coordinates) {
       alert('You clicked: ' +
               coordinates.pixel.x + ' x ' + coordinates.pixel.y +
               ' (Pixel), which equals ' +
               coordinates.world.x + ' x ' + coordinates.world.y +
               ' (World).');
   }
```

Angepassung der Funktion mapClickWorker() zum Aufruf einer URL

Alternativ kann beispielsweise ein neues Fenster mit einer URL geöffnet und die Koordinaten als Parameter übergeben werden. So können Sie beispielsweise OpenStreetMap aufrufen und die Koordinate des Klickereignisses zentrieren:

`https://www.openstreetmap.org/export#map=15/50.7311/7.0985`

```
_mapClickWorker: function(coordinates) {
       window.open('http://www.openstreetmap.org/export#map=15/' + coordinates.world.y + '/' +
↪coordinates.world.x);
   }
 src/Workshop/DemoBundle/WorkshopDemoBundle.php
```

CHAPTER

NINE

VERSIONSHISTORIE

English version of this document.

Note: **Hinweis:** Diese Versionshistorie wird nicht mehr gepflegt. Auf GitHub finden Sie den aktuellen Changelog und alle Versionen-Milestones.

9.1 Version 3.2.9

Release Datum: 28.01.2022

Verbesserungen und Bugfixes:

- https://github.com/mapbender/mapbender-starter/blob/master/CHANGELOG.md#v329

9.2 Version 3.2.8

Release Datum: 02.11.2021

Verbesserungen und Bugfixes:

- https://github.com/mapbender/mapbender-starter/blob/master/CHANGELOG.md#v328

9.3 Version 3.2.7

Release Datum: 07.09.2021

Verbesserungen und Bugfixes:

- https://github.com/mapbender/mapbender-starter/blob/master/CHANGELOG.md#v327

9.4 Version 3.2.6

Release Datum: 09.08.2021

Verbesserungen und Bugfixes:

- https://github.com/mapbender/mapbender-starter/blob/master/CHANGELOG.md#v326

9.5 Version 3.2.5

Release Datum: 08.06.2021

Verbesserungen und Bugfixes:

- https://github.com/mapbender/mapbender-starter/blob/master/CHANGELOG.md#v325

9.6 Version 3.2.4

Release Datum: 04.03.2021

Verbesserungen und Bugfixes:

- https://github.com/mapbender/mapbender-starter/blob/master/CHANGELOG.md#v324

9.7 Version 3.2.3

Release Datum: 21.12.2020

Verbesserungen und Bugfixes:

- https://github.com/mapbender/mapbender-starter/blob/master/CHANGELOG.md#v323

9.8 Version 3.2.2

Release Datum: 02.11.2020

Verbesserungen und Bugfixes:

- https://github.com/mapbender/mapbender-starter/blob/master/CHANGELOG.md#v322

9.9 Version 3.2.1

Release Datum: 06.08.2020

Verbesserungen und Bugfixes:

- https://github.com/mapbender/mapbender-starter/blob/master/CHANGELOG.md#v321

9.10 Version 3.2.0

Release Datum: 29.07.2020

Verbesserungen und Bugfixes:

- https://github.com/mapbender/mapbender-starter/blob/master/CHANGELOG.md#v320

9.11 Version 3.0.8.6

Release Datum: 15.09.2020

Verbesserungen und Bugfixes:

- https://github.com/mapbender/mapbender-starter/blob/master/CHANGELOG.md#v3086

9.12 Version 3.0.8.5

Release Datum: 05.02.2020

Verbesserungen und Bugfixes:

- https://github.com/mapbender/mapbender-starter/blob/master/CHANGELOG.md#v3085

9.13 Version 3.0.8.4

Release Datum: 04.09.2019

Verbesserungen und Bugfixes:

- https://github.com/mapbender/mapbender-starter/blob/master/CHANGELOG.md#v3084

9.14 Version 3.0.8.3

Release Datum: 05.07.2019

Verbesserungen und Bugfixes:

- https://github.com/mapbender/mapbender-starter/blob/master/CHANGELOG.md#v3083

9.15 Version 3.0.8.2.1

Release Datum: 05.07.2019

Verbesserungen und Bugfixes:

- https://github.com/mapbender/mapbender-starter/blob/master/CHANGELOG.md#v30821

9.16 Version 3.0.8.2

Release Datum: 03.07.2019

Verbesserungen und Bugfixes:

- https://github.com/mapbender/mapbender-starter/blob/master/CHANGELOG.md#v3082

9.17 Version 3.0.8.1

Release Datum: 14.05.2019

Verbesserungen und Bugfixes:

- https://github.com/mapbender/mapbender-starter/blob/master/CHANGELOG.md#v3081

9.18 Version 3.0.8.0

Release Datum: 12.04.2019

Verbesserungen und Bugfixes:

- https://github.com/mapbender/mapbender-starter/blob/master/CHANGELOG.md#v308

Hinweise zur Aktualisierung

https://github.com/mapbender/mapbender/blob/master/UPGRADING.md#308

9.19 Version 3.0.7.7

Release Datum: 07.11.2018

Verbesserungen und Bugfixes:

- https://github.com/mapbender/mapbender-starter/blob/master/CHANGELOG.md#v3077

9.20 Version 3.0.7.6

Release Datum: 18.10.2018

Verbesserungen und Bugfixes:

- https://github.com/mapbender/mapbender-starter/blob/master/CHANGELOG.md#v3076

9.21 Version 3.0.7.5

Release Datum: 26.09.2018

Verbesserungen und Bugfixes:

- Die Beschreibung zu den Fixes ist in den Repository-Links zu finden
- Aktualisierung Mapbender-Starter v3.0.7.5
- Aktualisierung Mapbender Repository auf v3.0.7.5
- Aktualisierung Owsproxy Repository auf v3.0.6.4, includes Owsproxy dependencies
- Aktualisierung mapbender/vis-ui.js Repository auf 0.0.72
- Aktualisierung mapbender/data-source Repository auf 0.1.8
- Aktualisierung mapbender/digitizer Repository auf 1.1.66
- Aktualisierung bundled Composer Repository auf 1.6.5
- ComposerBootstrap Bereinigungen

9.22 Version 3.0.7.4

Release Datum: 29.08.2018

Verbesserungen und Bugfixes:

- [Sicherheit] In Entwicklungsumgebungen kann es bei den Assets zu einem XSS Fehler kommen. Dieser Fehler tritt nur in einigen Umgebungen mit spezifischen PHP Einstellungen (error_reporting z.B. E_ALL) auf.
- Zurücknahme der Keyword Spalte zum Datentyp "varchar", um Inkompatibilitäten mit Oracle zu vermeiden. Zu lange Keywords werden auf 255 Zeichen abgeschnitten. (#1000)
- Einige JavaScript Fixes, die beim stillgelegten Internet Explorer 11 zu Problemen führten. (#990)
- Leere Layernamen werden beim FeatureInfo nicht mehr angefragt (PR #1010).
- Doctrine Optimierungen um die Layerreihenfolge Einstellungen in PostgreSQL zu setzen.
- Regressions-Fix beim WmsLoader und image format / info format Einstellungen.
- Fix beim Delete Cascade SQL Statement in PostgreSQL, wenn eine Wms Quelle gelöscht wird.
- Fix bei Übersetzungen, wenn nur ein Platzhalter ausgegeben worden ist. Diese nehmen nun die Fallback Übersetzung (per default: Englisch)
- OSGeo Logo aktualisiert (PR #861)

Anmerkungen zum Update:

Bitte führen sie wieder ein **app/console doctrine:schema:update** durch, um die Keyword-Tabelle wieder zu varchar zu ändern.

```
app/console doctrine:schema:update
```

Falls der Update Befehl fehlschlägt, z.B. mit der PostgreSQL Meldung `SQLSTATE[22001]: String data, right truncated:` und `7 FEHLER: Wert zu lang für Typ character varying(255)`, dann haben Sie einen Schlüsselwort-Eintrag in der Tabelle `mb_core_keyword`, der 255 Zeichen überschreitet. Diesen können Sie mit folgendem SQL-Statement herausfinden:

```
SELECT x, id, length(x) FROM (
  select value, id from  mb_core_keyword
) AS t(x) order by length desc;
```

9.23 Version 3.0.7.3

Release Datum: 13.07.2018

Allgemein:

- Ändern des Mapbender Logos und des Namens: Mapbender3 wurde der Einfachheit halber in Mapbender umbenannt und wir haben sowohl die Texte in der Dokumentation als auch die Logos augetauscht. Unsere URLs hatten wir schon auf http://mapbender.org umgestellt.
- Mapbender erwartet mindestens PHP > 5.6 zum laufenden Betrieb. PHP 7 wird empfohlen.

Neue Funktionen:

- QGIS Server Layerreihenfolgende, dokumentiert unter der Rubrik *Layerset*
- Neues Element: *Koordinaten Utility*
- Mouse-Over im SearchRouter
- GPS Button im POI
- Dynamisches Laden von Legendenbildern im Legenden-Element (PR #605, PR #606)
- Anzeige eines Cookie Banners in Anwendungen. Siehe *Cookie-Banner*.

Änderungen:

- Die Standard-Anwendungen befinden sich nun im Verzeichnis *(application)/app/config/applications*, jede in ihrer eigenen Datei. Dazu gehören:
  - Die Mapbender Demo Map Anwendung
  - Die Mapbender Demo Map basic Anwendung
  - Die Mapbender mobile Anwendung

Weitere YAML-Anwendungen können dort hinterlegt werden.

Verbesserungen:

WMS Dienste und WMS Loader:

- Verbesserungen im WMS Loader und der Service Kompatibilität, dessen Logik nun dem Backend folgt
- Fix im GetLegendGraphic Request auf geschützten Diensten über den Tunnel
- Fix und Verbesserungen im URL-Signen (#590)
- Viele Verbesserungen im WMS Backend
- Fix im Instance-Tunnel zum Zugriff auf geschützte Dienste
- Fix beim Zugriff auf WMS Dienste mit undefinierten Kontaktinformationen
- Verschiedene Fixes bei Min / Max Scale Definitionen in Sublayern gegenüber dem Root-Layer
- Fix der Layerreihenfolge in PostgreSQL

- Über WMS Loader zugefügte WMS Dienste und Metadaten Anzeige. Wir können die Metadaten nicht anzeigen, werfen aber keinen Fehler mehr

Design und CSS:

- Transparenz in der Zoombar und der Toolbar angepasst für eine gleichbleibende Farbe der Schaltflächen
- Fix im Erstellen von Anwendungen und dem Hinzufügen eines Bildes

Druck: * Fix im Druck von PNG8 Karten, wenn das Image Format "image/png; mode=8bit" ist * Fix im Druck bei speziellen Schriftgrößen (insbesondere unter Windows bei PHP 7.1) * Fix im Druck, wenn PHP-Notices angeschaltet sind und die yStartPosition fehlte (#555)

FOM:

- Verbesserung im FOM: Wrong Type Definition in ACL Provider Constructor #641
- Verbesserung im FOM beim SSPI

Übersetzungen:

- Verbesserung in den Übersetzungen. Danke an den Code-Sprint der FOSS4G!
- Änderung der Übersetzungen von XLIF nach YAML in den Modulen FOM und OWSProxy

Verschiedenes:

- Standardmäßig ist nun der "maximum feature count" Parameter für GetFeatureInfo Anfragen auf 1000 gesetzt
- Fix im Scale-Selector, der sich in einigen Fällen nicht aktualisieren wollte
- Fix im Aufruf von Mapbender mit POI-Parameter (#642)
- Fix im Legendenelement bei überlangen Legendenbildern (#640)
- Fix im Backend beim Hinzufügen von Elementen
- Fix eines Foreign Key Violation Fehlers in PostgreSQL, wenn einen Datenquelle gelöscht wird (PR#840)
- Einbau von Cookieconsent für Mapbender
- Dateiname Präfix für Druckausgaben geändert.

Code-Verbesserungen:

- Update auf Symfony 2.8 (siehe PHP Voraussetzungen)
- Einbau des Doctrine Migrations Frameworks
- Fix eines möglichen URL Signing Spoofings bei fehlenden URL Query Parametern
- Doctrine Param Coverter Definitionen (PR #645)
- WMSLayerSource: getAuthority (PR #542)
- DimensionsHandler (#610). Dieser kann in den kommenden Versionen veröffentlicht werden.
- Das Hinzufügen von Elementen im Backend konnte zu einem Fehler "Warning: usort(): Array was modified …" führen (#586)
- Element Template und AdminType Verbesserungen (#743)
- Serialisierung der MetadataURL (#747)
- UnitTest und deren Pre-Conditions (#760)

- USort und array_multisort durch einen PHP-Bug (#586)
- Strikte SCSS Warnungen beim Kompilieren mit ruby-sass
- Fix des Wachsens in "autority" bei wiederholtem Export / Reimport / Kopieren von Anwendungen
- Bypass bei (womöglich sehr langen) WMS Loader DTD/XSD Validationen von GetCapabilities Dokumenten
- PHP 5.6 Kompatibilität bei Migrations

Dokumentation:

- Neues Design der Dokumentation. Wir haben diese auf das Sphinx RTD Thema umgestellt, so dass die Dokumentation auch unter mobilen Geräten leichter lesbar ist. Ein Ausdruck der einzelnen Seiten ist möglich.
- Umstrukturierung der Dokumentation: Die einzelnen *Elemente* sind unterteilt in:
  - *Basis-Elemente*
  - *Suchen*
  - *Export / Druck*
  - *Editierung*
  - WMC
  - *Backend*
  - FOM
  - *Verschiedenes*
- Verbesserte Dokumentation zu den Elementen:
  - *Hintergrund wechseln*
  - *Button*
  - *Koordinatenanzeige (Coordinates Display)*
  - *HTML (Element)*
  - *Legende (Legend)*
  - *Karte (Map)*
  - *Navigationswerkzeug (Navigation Toolbar)*
  - *Übersichtskarte (Overview)*
  - *Suchen (Search Router)*
  - *SRS Auswahl (SRS Selector)*
- Dokumentation: Erweiterung zu dem printclient und den neuen dynamischen Features bei den Drucktemplates.
- Dokumentation: Aufnahme des MS4W Pakets zur Installation unter *Installation unter Windows*. Vielen Dank an Jeff McKenna.

Anmerkungen zum Update:

Bitte führen sie ein **app/console doctrine:schema:update** durch, wenn Sie auf diese Version aktualisieren. Die QGIS-Layerreihenfolge benötigt eine Änderung in der Mapbender-Datenbank. Auch die 255 Zeichen für WMS-Dienste erforderten eine Änderung der Datenbank.

```
app/console doctrine:schema:update
```

9.24 Version 3.0.7.2, 3.0.7.1 und 3.0.7.0

Aufgrund von Tagging-Fehlern im Code auf Github wurden diese Versionen nie offiziell veröffentlicht. Da es nicht korrekt ist, Code zu re-taggen, führen wir die Entwicklungslinie mit Version 3.0.7.3 fort.

9.25 Version 3.0.6.3

Release Datum: 27.07.2017

Bugfixes:

- Regression: Koordinaten-Reihenfolge bei Anfragen an WMS 1.3.0 gefixt. Koordinatensysteme mit getauschter Axis-Orientation werden von der Karte, Druck und Export unterstützt. (#529)
- Regression: ScaleHint bei WMS Diensten korrigiert. Einige WMS Dienste mit einem Scale in den Capabilities konnten nicht in die Anwendung eingeladen werden. (#584)

9.26 Version 3.0.6.2

Release Datum: 20.07.2017

Bugfixes:

- Search Router: Fehler mit OpenLayers gefixt (#543)
- Search Router: Auto Close nach dem Klick in der mobilen Anwendung (#548)
- Koordinaten-Reihenfolge bei Anfragen an WMS 1.3.0 gefixt (#529)
- Druck: Darstellung von Punkten und Labels bei hochauflösendem Druck (#573, #574, #492)
- Abspeichern von WMC im WMC Editor Dialog (#577)
- ScaleHint bei Sublayern von 1:1 korrigiert (#565)
- Verbreiterung der Titel-Spalte bei Layerset-Instanzen (#559)
- Kommando, um den Imagepfad in bestehenden Map-Elementen zu aktualisieren (#530)
- Übersetzung Drucken Schaltfläche im FeatureInfo Dialog (#552)
- Anpassungen Default-Wert “immediate” bei Messtool (#538)
- SRS: Definitionen aktualisiert (#550, #562) und YAML-Standard-Anwendungen angepasst (#561)
- Anpassungen an der Doku bei den Translations.

Zusätzliche Update Schritte:

(1) Aktualisierung der EPSG-Codes

Führen Sie nochmals den Befehl `app/console doctrine:fixtures:load --fixtures=mapbender/src/Mapbender/CoreBundle/DataFixtures/ORM/Epsg/ --append` aus. Es werden zwei neue Koordinatensysteme (EPSG:4839|ETRS89 / LCC Germany (N-E) und EPSG:5243|ETRS89 / LCC Germany (E-N)) der Mapbender Tabelle `mb_core_srs` hinzugefügt.

(2) Aktualisierung der Parameter im Map-Control

Führen Sie den Befehl `app/console mapbender:upgrade:database` aus, um den OL-ImagePath Parameter des Map-Controls anzupassen: Von `bundles/mapbendercore/mapquery/lib/openlayers/img` nach `components/mapquery/lib/openlayers/img`. Das ist notwendig, falls Sie das POI-Element nutzen bzw. Mapbender mit dem poi-Parameter aufrufen und keine Sprechblase für den POI sehen. Beispiel: https://demo.mapbender.org/application/mapbender_user?poi{[}point{]}=366164%2C5623183&poi{[}scale{]}=25000&poi{[}label{]}=Please+take+a+look+at+this+POI%3A

9.27 Version 3.0.6.1

Release Datum: 24.05.2017

Bugfixes:

- Druck zeigte falsche Maßstäbe im Kartendisplay.
- Spezifische WMS konnten nicht gedruckt werden, wenn sie als HTTP Antwort image/pntg; charset-iso… übertragen haben.
- Backend: FOM Dialoge mit vielen Einträgen machten die Checkboxen unbenutzbar.
- config.php wieder zurück im mapbender-starter.
- Aktualisierung des bin/composer Kommandos zum Bauen von Mapbender.
- Style-Fix bei Administrations-Dialog des Basesource-Switcher.
- WmcEditor Standard Parameter für Breite und Höhe eingetragen.
- Aktualisierung Startseite Dokumentation.
- Kleine Styling Verbesserungen im Backend.
- Kleine Säuberungen des Code.

9.28 Version 3.0.6.0

Release Datum: 05.05.2017

Architektur:

- Systemvoraussetzung PHP: 5.5.4 oder höher
- Unterstützung von PHP 7.
- Mapbender, FOM and OWSProxy als Module ausgelagert. Sie sind nun in der composer.json eingebunden.
- Die Dokumentation ist Teil des Composers.
- Anpassungen ElementGenerator
- Ermitteln von Benutzerrollen
- Composer Einträge mit https
- Verschiedene Verbesserungen an Controllern und Bundles.
- Doctrine generate Kommandos als deprecated markiert
- Doctrine assets:dump Kommando als deprecated markiert

- Aktualisierung JOII Bibliothek
- Ablage von Symlinks zu verschiedenen Binaries im bin Verzeichnis
- Composer abgelegt im application/bin Verzeichnis
- Check in der Composer-Installation, ob die SASS Compiler Binaries ausführbar sind. Falls nicht, werden sie ausführbar gesetzt.
- Neue Composer-Kommandos zum Generieren der Dokumentation: Nur API Dokumentation generieren: bin/composer gen-api-docs
- Neue Composer-Kommandos zum Generieren der Dokumentation: Nur Benutzerdokumentation generieren: bin/composer gen-user-docs
- Verweis auf eigene Forks von open-sans, joii, compass-mixins und codemirror Bibliotheken.

Bugfixes und Features:

- Das Messen zeigt die Koordinaten live an, d.h. beim Bewegen der Maus werden schon jeweils die erreichten Segment- und Gesamtlängen angezeigt.
- Die Messergebnisse werden nun von oben hinzugefügt. Damit steht das aktuelle Messergebnis sichtbar an oberster Stelle und muss nicht gescrollt werden.
- Der Login Dialog (Registrierung, Password vergessen) wurde für mobile Geräte optimiert, um den Zugriff auf gesicherte mobile Anwendungen zu erleichtern.
- Neu hinzugefügte Layerset Instances sind nun nicht mehr per Default als Basesource markiert.
- Das Copyright Element Popup kann mit einer Höhe und Breite definiert werden.
- Das Löschen von Layersets führte in einigen Fällen zu einem korrupten Map-Element und falschem Layertree
- Anpassungen und Vereinfachung der Styles des FullScreenTemplate
- Einführung der Prüfung der CSS Angaben in der Anwendungskonfiguration
- Delay beim Wechsel von Layern gefixt
- Gefixter GetMap Request bei veränderter Reihenfolge der Layer im TOC
- Fix für WMS support 1.3.0
- Fix für geschützte WMS Dienste bei GetMap, GetFeatureInfo, Print, Export und Legende
- Fix für geschützte WMS Dienste, bei denen im Username oder Passwort ein Hashzeichen vorkommt.
- Fix des WMS Parameters Exception Format bei GetMap und GetFeatureInfo Request (Github-Issue 400)
- Fixes bei den Layer-Styles im GetMap und GetFeatureInfo Request
- Default Tile Size bei der Kartenkomponente auf 512 gesetzt (war 256)
- WMS Keyword Begrenzung (war: 255 Zeichen) ist geändert. Der Spaltentyp wurde auf “text” geändert. Das Kommando app/console doctrine:schema:update ist notwendig, um die MB3-Datenbank einer Vorversion zu aktualisieren,
- Fix beim Import von YAML Anwendungen und der Erstellung von Duplikaten bei WMS Datenquellen.
- Fix bei Minimal und Maximal Scale Hint Angaben eines WMS Dienstes.
- Druck: Farbe kann bei variablen Texten eingefügt werden.
- Druck: Druck der Legende, wenn der Dienst über einen Proxy eingebaut wird.

- Druck: Dienste, die mit PNG8 registriert wurden, konnten in einigen Fällen nicht exportiert oder gedruckt werden. Die Typen image/png8 und image/png mode=24bit werden nun unterstützt.
- Druck: In einigen Fällen wurde die Legende nicht ausgegeben, wenn OWSProxy aktiviert war
- BaseSourceSwitcher: Mehrere Requests an einen WMS der auf nicht sichtbar gesetzt war, wenn der BaseSourceSwitcher als Menü angezeigt wurde.
- Überflüssige WMS Anfragen an bestimmte WMS, abhängig vom Maßstab.
- Ablage von YML-Anwendungen in application/app/config/applications: mapbender_mobile.yml, mapbender_user_basic.yml, mapbender_user.yml und Anpassungen in der WMS Version
- Administrationsoberfläche YAML Editierung und Umformatierungen nach dem Speichern (Github-Issue 350)
- Anpassungen POI-Koordinate: Transformation und SRS, Nachkommastellen
- Fix eines XSS Fehlers im POI Dialog
- Fix in POI Dialog, wenn useMailto = false
- GPS: Fehlermeldungen, wenn kein Signal und Abhängigkeit von Chrome-Browsern und https.
- GPS: Verschieben der Karte nur bei erster Positionsbestimmung.
- User-Interface: Scrollen der Dropdownliste im Backend, z.B. bei den Icons für Buttons, scrollte auch den Hintergrund.
- "Only valid" Checkbox beim Einladen eines WMS ist nun standardmäßig nicht mehr aktiviert.
- Umformatierte Meldungen, wenn die Schemata eines WMS beim Hinzufügen nicht zugreifbar sind.
- Der SearchRouter zeigt, wenn er in der Sidepane eingebettet ist, die Schaltflächen Suche und Reset.
- Internet Explorer Kompatibilität: Anpassungen Zoombar.
- Internet Explorer Kompatibilität: Anpassungen OverviewMap.
- MS Egde Kompatibilität: Import Dialog (https://connect.microsoft.com/IE/feedback/details/1574105/microsoft-edge-file-upload-bug-build-10240-rtm)
- Verbesserung der Performance bei *einigen* Installation unter Windows durch WinCachePHP und PHP Opcache (für Details siehe Installation unter Windows)
- Änderung der Systemvoraussetzungen: Für Windows ist die "Non-Thread-Safe" Variante von PHP notwendig!
- Kopieren von Anwendungen für Benutzer, die nicht root sind (ACL Application: owner, Benutzer: owner, Berechtigungen ACLs: owner, Element: owner, Gruppen: owner, Service Source: owner, spezifische Anwendung: owner)

FOM und Absicherung:

- Anzeige von Benutzern, die Zugriff auf ein Element in der Anwendung haben.
- Überarbeitung des Secure Elements Dialogs
- Benutzer mit der Rolle View bei Diensten dürfen Metadaten sehen und Dienste in eine Anwendung einladen.

verschiedenes

- Anpassung Design FeatureInfo bei Anzeige als Accordion und wenn nicht als Source angezeigt
- Verschieben von Popups innerhalb einer Anwendung
- Anpassungen WMC Edit Dialoggröße und XSS

- Fix in den Übersetzungen
- YAML basierende Anwendungen können die Sidebar anpassen: align (left/right), closed (true/false), width (px/em/%)
- Backend: Target-Feld: Leere Auswahl bei Drop-Down Feldern.
- Anpassungen WMS Scale, ScaleHint und Min/Max Werte beim Öffnen einer Layerset-Instance
- Anzeige des WMS-Titels in den Metadaten des TOC bei aktualisiertem WMS
- Anzeige des Applikations-Logos in der Konfiguration
- Anpassungen Simple Search und SearchRouter
- Fehlermeldung bei falsch konfigurierter Layerset-Instance
- Druck: Einführung von setasign/pdf als Alternative zu toooni/fpdf
- Druck: Fix der Fehlermeldung bei fehlendem Drucktemplate
- Messen von Linien und Flächen in WGS84 (EPSG:4326)
- Anpassung Anzeige der Element ACL
- WMS Aktualisierung: Entfernen von User/Passwort Browser Autocomplete
- Anzeige der Nachkommastellen im Coordinates-Display
- Anpassungen, Erweiterungen EPSG import
- Layer Maßstäbe und Fix der Default Visibility eines Layers
- Entfernen des Data Source Monitor Icon (kommt in Version “Next”)
- Administration: Bewegung über Tabulator-Taste
- Verbesserung der Konfigurationsoberfläche
- Anzeige der Source-ID in Anwendungen
- Verbesserung Caching Mechanismen
- Verbesserungen Export / Kopieren
- Vereinfachung bei der Erstellung neuer Elemente
- Entfernen des provide ext-ldap Statements in Composer. Die Komponenten wurden ausgelagert. Wir werden die LDAP Module in Version 3.0.7 einbauen.
- Restrukturierung von DataManager und DataSource seit der Version 1.0.2 des data-manager.

Mobiles Template

- Generelle Verbesserungen des Mobilen Templates
- Fix für einen Button, wenn dieser nur auf einen Link verweist.
- Fix für das Icon Label und der Schriftdicke normal
- Verbesserungen des Button Handlings allgemein
- Fix des event handlers “”on moveend”

Digitizer

- Update Digitizer auf Version 1.1.
- Druck von Multipolygonen.
- Objekte erscheinen nicht mehr im Druck, wenn Sie im Digitizer ausgeblendet worden sind.

- MinScale Einschränkung hinzugefügt
- Objekte mit einer Linienbreite von 0 werden im Druck nicht mehr sichtbar.
- Anpassungen der Close Schaltfläche: “allowCancelButton” und “allowDeleteByCancelNewGeometry”.

Form Generator:

- Anpassungen: Hinzufügen des HTMLElement handling für Services und DataStore Konfiguration.

Dokumentation

- FAQ der Dokumentation hinzugefügt.
- Einführung der Contributing Guide für Mapbender-Starter und OWSProxy. Mapbender selbst und FOM werden folgen. Dies sind die Einstiegsdokumentationen für Entwickler und Mitwirkenden von Mapbender.
- Die Developer Dokumentation wird dort aktualisiert und aus dieser Benutzerdokumentation schrittweise überführt. Somit wird in der Zukunft diese Dokumentation hier sich mehr an die Anwender richten, während die Entwickler ihre Dokumentation direkt im Source-Code der einzelnen Module finden.
- Ausführlichere Layertree Dokumentation

config.yml Anpassungen

DBAL-Parameter:

- default_connection: Bei mehreren Einträgen definiert die default_connection, die Standardverbindung der MB3-Datenbank.
- persistent: Persistente Verbindungen zur Datenbank zwecks Performance (Oracle)

```
doctrine:
  dbal:
    default_connection: default
      connections:
        default:
          ...
          persistent: true
```

mapbender-starter/application/app/config/applications/

Verzeichnis, in das YAML-basierende Anwendungen abgelegt werden können. Als Beispiel sind die drei bekannten Anwendungen Mapbender-User, Mapbeder-User-Basic und Mapbender-Mobile abgelegt.

app/console doctrine:schema:update

```
app/console doctrine:schema:update --dump-sql
ALTER TABLE mb_core_keyword ALTER value TYPE TEXT;
ALTER TABLE mb_core_keyword ALTER value DROP DEFAULT;
```

9.29 Version 3.0.5.3

Release Datum: 04.02.2016

Bugfixes:

Besondere Änderungen:

- Performance: Die CSS, JavaScript und Translation Dateien werden nun im Produktionsmodus im Symfony Cache gehalten. Dies kann insbesondere bei langsamen Servern zu Performancesteigerungen führen. Die Änderung gilt nicht für den Entwicklungsmodus (app_dev.php).
- Das Paket eslider/sassc-binaries bietet nun auch einen sassc Compiler für 32-bit Linux Systeme an. Dies führte zu Anzeigeproblemen unter 32-bit Linux Systemen (http://lists.osgeo.org/pipermail/mapbender_users/2015-December/004768.html)
- Redlining: Die Inhalte des Redlining Elements sind nun sichtbar und das Element kann sowohl als Dialog als auch Element in der Sidepane verwendet werden. Siehe auch die Dokumentation zum Redlining Element. Die Scrollbar bei den Geometrie-Types im Konfigurationsdialog wird korrekt angezeigt.

Benutzer und Absicherung:

- Benutzer können durch Administratoren, die mindestens das ACL-Benutzerrecht "Operator" besitzen, aktiv oder inaktiv geschaltet werden. Das ist z.B. sinnvoll, wenn Benutzer, die sich selbst registriert sich selbst noch nicht aktiviert haben. Siehe die Dokumentation zu Benutzern.
- Texte, Übersetzungen und Styles bei der Registrierung und dem Passwort Reset wurden angepasst, ebenso die Dokumentation dazu

Druck und Export Image:

- Das Print-Modul kann nun auch in die Sidepane eingebaut werden.
- Print Legende: Die Größe der Legende im Ausdruck wurde verkleinert, um die Anzeige zu verbessern.
- Druckvorlagen: Die Standard-Druckvorlagen sind angepasst worden. Der Abstand der dynamischen Texte wurde verringert und die Ausrichtung verbessert.
- Druck: In der Druck-Konfiguration wurden in Kombination von notwendigen (required: true) und optionalen Feldern (required: false) die optionalen Felder teilweise als required angezeigt (Github #380).
- Mapbender druckte unter Umständen die Legende aller Layer eines WMS aus, auch wenn der Layer nicht aktiv war (festgestellt im Mapbender_Users WMS).
- Export Image: Die Transparenz von gekachelten und nicht-gekachelten Diensten wird im Export Image unterstützt.

Kopieren und Import:

- Kopieren einer Anwendung unter SQLite und MySQL: Es lag ein Fehler vor, bei dem Anwendungen nicht kopiert werden konnten, wenn das Mapbender3 Repository in einer SQLite oder einer MySQL Datenbank lag.
- Fehler beim Import von Anwendungen als JSON unter MySQL (Elemente verlieren ihr Target) wurde gefixt.

Einzelne Elemente:

- **WMC** und Thematische Layer: Wenn ein WMC geladen wird und Keep Sources auf "no" eingestellt ist, werden auch die Thematischen Ebenen aus dem Layerbaum genommen.

- **WMS-URL Parameter** und Legende: Wenn ein Dienst über den wms_url Parameter in die Anwendung geladen wird, wurde die komplette Legende angezeigt und nicht die für die jeweiligen Layer. Dieses Verhalten wurde gefixt.
  - *Hinweis*: Es gibt WMS Dienste, die eine Legende im Hauptlayer-Element definieren. Diese wird nach der WMS Spezifikation auf die Layer vererbt, die selbst keine Legende definiert haben (z.B. weil sie nur eine Schrift anzeigen). Der Effekt in MB3 ist ähnlich, die Ursache aber eine andere, so dass in diesem Fall Anpassungen an den Capabilities notwendig sind (für den Layer eine statische Legende definieren).
- **Thematische Layer**: Korrektur im An- und Abschalten von Layern, die in einem eigenen Layerset sind, aber nicht als Thematischer Layer angezeigt werden.
- **Koordinatendisplay**: Das Element zur Koordinatenanzeige zeigte "null" als Text für den Präfix oder Separator, obwohl diese Felder leer sein sollten. Das Element hat weiterhin eine feste Breite bekommen, so dass das Layout im Footer stabiler wird. Der Wert kann verändert werden (Siehe die CSS-Anpassungen zu dem Element).
- **SearchRouter**: Der Inhalt der Ergebnisse füllt nun das gesamte Dialogfenster aus und passt sich an die Größenänderung an. In der Sidebar wird die gesamte Höhe ausgenutzt. Der SearchRouter ist, wenn er als Dialog eingesetzt wird, in der Höhe und Breite konfigurierbar.
- **ScaleSelector**: Die Breite des Elements kann mit einem CSS-Statement verändert werden und ist nicht auf 155 Pixel festgelegt.
- Wenn in einer **Layerset-Instance** alle Layer auf visible=off gestellt sind, waren der Tree und die Legende nicht sichtbar.
- **POI Dialog** Verbesserungen im Styling, wenn usemailto auf false gesetzt ist.
- **Layertree**: Anzeige der Titel nun mit einer Gesamtlänge von 40. Der Default-Wert wurde verändert. Sie können den Parameter Titlemaxlength anpassen.
- **GPS**: Verfeinerungen im GPS Tool

Allgemeine Änderungen:

- Bei Änderungen an Grunddaten einer Anwendung, dem Layout, den Layersets, dem CSS und der Sicherheit bleibt man nun in dem jeweiligen Reiter und springt nicht zu den Grunddaten zurück.
- Allgemeine Verbesserungen im Digitizer Code Version 1.0. Die Version 1.1 ist kompatibel mit Mapbender 3.0.5.3.
- Github Infodateien: Kleine Aufräumarbeiten im Github Repository, um die automatischen Buildprozesse zu verbessern.
- Transparenz von Diensten: Dienste, die mit einer Transparenz dargestellt werden aktualisierten sich mit einem unschönen Effekt, verursacht durch durch den "transitionEffect" in OpenLayers. Dieser Effekt wurde entfernt.
- Gruppenfilter: In dem Konfigurationsdialog zu den Sicherheitseinstellungen wurde die Auswahl von Gruppen verbessert, wenn Gruppen einen gleichem Namen aber unterschiedlichen Suffixe besitzen.
- TileSize Parameter in der Kartenkonfiguration wird unter Umständen nicht gesetzt.
- Anzeige der Symbole unter Internet Explorer 11 und MS Edge 25 (auch ein Fehler in MS Edge 20).
- mapbender.yml: Beim initialen Import der mapbender.yml werden die Angaben von GetFeatureInfo nun auf text/html gesetzt. Die mapbender.yml kann nun mit Redlining erweitert werden.

Umzug der Mapbender Domänen:

- Wir haben die URL www.mapbender.org auf die Mapbender3 umgeschwenkt. Somit ist die Mapbender3 Seite zukünftig über www.mapbender.org und www.mapbender3.org erreichbar. Mapbender2 ist nun über www.mapbender2.org erreichbar.
  - http://www.mapbender.org: Mapbender3,
  - http://www.mapbender3.org: Mapbender3,
  - http://www.mapbender2.org: Mapbender2.

Bekannte Probleme:

- Das Sketch Tool funktioniert nicht korrekt und wird in Zukunft in das Redlining Tool übernommen werden.
- Karte weiterempfehlen funktioniert nicht für Facebook, Twitter und Google+.

9.30 Version 3.0.5.2

Release Datum: 27.10.2015

Bugfixes:

- Kopieren von Anwendungen: Rechte und Gruppen werden mit übernommen. Der Nutzer, der die Anwendung kopiert hat, wird Owner der kopierten Anwendung.
- FOM: Änderungen im Verhalten bei falschen Logins und dem Locking. Es wird nur noch angezeigt, dass der Login fehlgeschlagen ist - unabhängig davon, ob der Benutzer existiert oder nicht.
- Korrigierte Fehlermeldung beim Anlegen eines Benutzers mit zu kurzem Passwort.
- Print: Fix des Replace Patterns.
- Print: Fix, wenn falsch konfigurierter WMS Sonderzeichen (%26) in der Legenden URL hat.
- Image Export in Firefox.
- WMC Loader: Einladen von WMC und das Verhalten der BaseSources.
- BaseSourceSwitcher: Kacheln des nicht sichtbaren Dienstes werden nicht vorgeladen.
- BaseSourceSwitcher: Wenn eine Gruppe definiert ist, ist nur ein Thema beim Start eingeschaltet.
- SearchRouter: Fix der Anführungszeichen für Tabellennamen.
- Anwendungen kopieren: Fix der Suche in der kopierten Anwendung.
- Simple Search: Return Schaltfläche wird abgefangen.
- FeatureInfo: Add WMS Funktionalität und WMS Loader.
- Icon Polygone in der Toolbar von Anwendungen ist sichtbar.
- Icons, die nicht auf FontAwesome basieren, funktionieren auch im Mobilen Template.
- Administration Map Element: Anzeige des Konfigurationdialogs im Backend beginnt oben.
- Administration Datenquelle: Keine Formulardaten Vorbelegung durch den Browser für Benutzername und Passwort.
- Mobile Anwendung: Darstellung unter Firefox für Android.
- Update 3.0.4.x: FeatureInfo autoopen=true bleibt erhalten.
- Doku: FOM UserBundle Übersetzung und Ergänzung bei falschen Benutzer-Logins.

- Doku: URL Parameter scale im Map Element.
- Doku: WMC Loader und KeepSources.

Änderungen in der config.yml:

- Die folgenden Änderungen sind optionale Parameter für das Verhalten des Logins (siehe das entsprechende Kapitel im FOM Bundle):

```
fom_user:

  # Allow to create user log table on the fly if the table doesn't exits.
  # Default: true
  auto_create_log_table: true

  # Time between to check login tries
  login_check_log_time: "-5 minutes"

  # Login attemps before delay starts
  login_attempts_before_delay: 3

  # Login delay after all attemps are failed
  login_delay_after_fail: 2 # Seconds
```

9.31 Version 3.0.5.1

Release Datum: 26.08.2015

Neue Funktionen: im Kartenelement und beim Druck Client:

- Map: OpenLayers TileSize: Es kann die Kachelgröße für die Karte angegeben werden. Default: 256x256.
- Map: Delay before Tiles: Für WMS-T, z.B. bei zeitlichen Parametern (zukünftig)
- Druck: Koordinatenanzeige in der PDF-Druckausgabe
- Druck: Übernahme Druckmaßstab abhängig vom Kartenmaßstab
- Druck: Druck legend_default_behaviour
- Druck: Hinzufügen von Druckvorlagen über das +-Symbol
- Druck: Benutzerabhängiges Logo und Text

Bugfixes:

- Layertree: Lade-Symbol und Ausrufezeichen-Symbol.
- Layertree: Zoom Symbol nicht bei Layern ohne BBOX-Information
- WMS Reload: FeatureInfo
- WMS Reload: Einige WMS konnten sich nicht neuladen lassen.
- Export/Import von Anwendungen verschiedene Bugfixes
- WMC-Editor und WMC-Load Fixes.
- WMC aus einer Mapbender 3.0.4.1 Anwendung
- Tile Puffer und BBOX Puffer fixes

- FeatureInfo: Fixes im Design und bei der Anzeige als Accordion Panel
- FeatureInfo: Drucken
- Falscher Link Jquery-UI in layerset instance
- Save Layerset und Save Layout bleibt auf der Seite
- Classic Template: SCSS korrigiert
- Mobile Template: Bootstrap Meldung verdeckt Schließen Schaltfläche
- Mobile Template: SearchRouter Fenster ausblenden
- Mobile Template: Mozilla Firefox Fixes im Layout
- Backend: Layerset Filter und +-Buttons verstecken nicht mehr alles
- composer.json Anpassung Version Digitizer auf 1.0.*
- Dokumentation des JS-UI Generators (Formular-Generator): https://github.com/eSlider/vis-ui.js
- Umstrukturierte Installations-Dokumentation und einige Anpassungen (php-pear, assets-Verzeichnis, init:acl, openssl).
- Verbesserte Dokumentation der Mapbender3 Templates
- Verbesserte Dokumentation des Quickstart

Known Issues:

- Kopieren von alten 3.0.4.1 Anwendungen kopiert nicht die anzuzeigenden Layersets der Map. Bitte speichern Sie vorher das Map und Overview-Element.
- Regional Template entfernt

9.32 Version 3.0.5.0

Release Datum: 01.07.2015

Übersicht der Änderungen finden Sie unter: https://github.com/mapbender/mapbender-starter/blob/develop/CHANGELOG.md

- **WMS neuladen:** WMS Quellen können nun neugeladen werden, wenn sich deren Strutkur geändert hat.
- **Digitalisierung:** Im Rahmen des Releases wurde das neue Element Digitizer eingeführt. Über dieses kann durch eine YAML-Definition eine Erfassungsmaske für Punkte, Linien oder Flächen aufgebaut werden. Dabei wird wie bisher PostgreSQL als Datenquelle unterstützt. Oracle und SpatiaLite sind experimentell verfügbar. Die Entwicklung wurde so durchgeführt, dass die Erfassung auch auf andere Datenquellen wie z.B. OGC WFS erweitert werden kann.
- **Druck mit Legende:** Im Druck ist es nun möglich, die Legende auf einer separaten Seite auszugeben. Die Ausgabe kann über eine Checkbox gesteuert werden.
- **Konfigurierbarer Layerbaum:** Der Layerbaum unterstützt nun mehr als ein layerset. Sie müssen das Kartenelement anpassen, um die Layersets festzulegen, die angezeigt werden sollen sowie den Layerbaum selbt. Die Dokumentation befindet sich unter auf der Seite zum Layertree.
- **Verbesserte Infoausgabe:** Die Ausgabe der Infoabfrage wurde für die neue Version verbessert. So bleiben nun die Stile der Infoabfrage erhalten. Dienste, die keine Antwort liefern, werden nicht über einen Reiter angezeigt. Es erfolgen Meldungen, wenn keine Antwort geliefert wurde.

- **Mobiles Template:** In mehreren Projektlösungen haben wir uns bereits mit einer mobilen Lösung auf Basis von Mapbender3 auseinandergesetzt. Nun wird diese Lösung als Mapbender Mobile Template in der neuen Version 3.0.5.0 zur Verfügung gestellt. Sie finden eine neue Demo-Anwendung in der mapbender.yml mit Namen Mapbender Mobile (mapbender_mobile). Diese können Sie als Vorlage für Ihre Lösung verwenden. In der Release-Demo kann die Anwendung „Mapbender Mobile" getestet werden.
- **SASS Compiler:** Änderungen an der Architektur bezüglich des SASS Compilers führen zu einer performanteren Oberfläche.
- **Vendor Specific Parameter:** Eine WMS Layer Instanz unterstützt nun die Angabe von Vendor Specific Parametern, die an einen WMS Request angehangen werden. Die Werte können fest vergeben werden oder auf die User- und Gruppeninformation des angemeldeten Benutzers zurückgreifen. Dokumentation ist unter dem Abschnitt Vendor Specific Parameters verfügbar.
- **Formular-Builder:** In Zusammenhang mit der Digitalisierung können für die Erfassung von dazugehörigen Sachdaten sehr komplexe Formulare generiert werden. Hierbei wurde sich an den Möglichkeiten, die in Mapbender 2.x zur Verfügung stehen, orientiert.
- **Neue Schaltflächen:** Einige Schaltflächen basieren auf einer neuen Schriftart, die alten Schaltflächen sind noch mit dem Namen FontAwesome verfügbar.
- **URL Parameter:** Mapbender3 kann mit Startparametern aufgerufen werden. Eine Liste der Parameter findet sich in der Dokumentation zu den URL Parametern.
- Neue Übersetzungen für Portugiesisch und Russisch.
- Symfony Update auf 2.3.30.

Änderungen in der config.yml:

- Änderung bei einer dbal connection:
  - **logging: false**: Die Option sorgt dafür, das *alle* SQL's nicht mehr geloggt werden. Mehr dazu hier: http://www.loremipsum.at/blog/doctrine-2-sql-profiler-in-debugleiste/
  - **profiling: false**: Profiling von SQL Anfragen. Diese Option kann in der Produktion ausgeschaltet werden.

    Wo möglich sollen die Optionen so umgestellt werden, dass die erst in Debug modus aktiv werden:

```
logging:              "%kernel.debug%"
profiling:            "%kernel.debug%"
```

Bekannte Probleme

- Beim Kopieren einer Anwendung von Mapbender 3.0.4.x muss in der Map/Overview der jeweilige Layerset neu gesetzt werden.

9.33 Version 3.0.4.1

Release Datum: 23-01-2015

Übersicht der Änderungen finden Sie unter: https://github.com/mapbender/mapbender-starter/blob/develop/CHANGELOG.md

- parameter 'layerRemove' removed from layertree configuration
- option 'removelayer' added into layertree menu

- container accordion structure changed
- import / export from applications added (without acls)
- display layer metadata
- Frontend: Sidepane Accordeon Legend is displayed without horizontal Scrollbar
- Backend: WMS Instanz configuration - contextmenu for layers shows wrong ID (only instance ID)
- Frontend: Legend - displays WMS Information although the checkbox Show
- Frontend: Layertree - contextmenu zoomlayer does not use the layer extent
- Backend: Add Source with user/password - informations is added to field originUrl not to fields user and password
- app/console mapbender:generate:element fixed errors
- bug visiblelayers fixed
- WMS with authentication saves in table mb_wms_wmssource username and password
- no metadata for applications coming from mapbender.yml definition (no entry in context menu)
- copy an application via button on application fixed
- print template resize northarrow, overview added
- improved screenshot for application handling
- https://github.com/mapbender/mapbender/milestones/3.0.4.1

9.34 Version 3.0.4.0

Release Datum: 12-09-2014 Übersicht der Änderungen finden Sie unter: https://github.com/mapbender/mapbender-starter/blob/develop/CHANGELOG.md

- Wechsel zur MIT Lizenz
- Symfony Update 2.3 LTS
- OpenLayers 2.13 mit zusätzlichen Patches
- Dienste Aktivieren über Button oder Menü (BaseSourceSwitcher)
- HTML-Element
- CSS-Editor für Anwendungen
- Reiterstruktur in der Seitenleiste
- Laden von Vorschaubildern für Anwendungen
- Import/Export von Anwendungen und Diensten
- spanische Übersetzung

9.35 Version 3.0.3

Release Datum: 17-03-2014 Übersicht der Tickets finden Sie unter: https://github.com/mapbender/mapbender/issues?milestone=8

- Erweiterungen Such-Router für SQL-Suchen (Selectboxen, Distinct)
- WMC Editor und Loader
- WMSLoader Erweiterung WMS über Link hinzufügen
- i18n - Internationalisation (english / german)
- Sketch zum Zeichnen von Skizzen
- POI - Treffpunktfunktion
- Bildexport zur Ausgabe von png und jpg
- WMS Anzeige über Button wechseln
- Druckausgabe mit Übersichtskarte, Replace-Pattern, optionalen Feldern
- Zusammenstellung von mehreren Elementen in der Seitenleiste (Wechsel über Button)
- Layerbaum mit Kontextmenü zur Transparenzeinstellung und zum Zoom auf das Thema
- Übergabe von Parametern beim Öffnen der Anwendung (Position)
- ACL für Elemente
- Funktion zur Validierung von WMS GetCapabilities Dokumenten

9.36 Version 3.0.2

Release Datum: 27-11-2013 Übersicht der Tickets finden Sie unter: https://github.com/mapbender/mapbender/issues?milestone=6

- Such-Router für SQL-Suchen
- WMC Editor und Loader
- WMSLoader Erweiterung WMS über Link hinzufügen

9.37 Version 3.0.1

Release Datum: 06-09-2013

Übersicht der Tickets finden Sie unter: https://github.com/mapbender/mapbender/issues?milestone=5

- Kopieren einer Anwendung mit Diensten
- Popup - draggable
- PrintClient Erweiterung Druck EPSG 4326, neue Drucklayouts, Druck A4-A0
- Abfangen von fehlerhaften Anmeldungen zum Abwenden von brute force login Versuchen
- Bug fixes

9.38 Version 3.0.0.2

Bugfix-Release Datum: 19-07-2013

Übersicht der Tickets finden Sie unter: https://github.com/mapbender/mapbender/issues?milestone=4

9.39 Version 3.0.0.1

Bugfix-Release Datum: 07-06-2013

Übersicht der Tickets finden Sie unter: https://github.com/mapbender/mapbender/issues?milestone=3

9.40 Version 3.0.0.0

Release Datum: 29-05-2013

Übersicht der Tickets finden Sie unter: https://github.com/mapbender/mapbender/issues?milestone=1

- Administrations Backend für Services, Applikationen, Benutzer/Gruppen und Zugriffsverwaltung
- Backend-/Frontend Design
- Zugriffsverwaltung
- Benutzer-/Gruppen-Administration
- WMS Administration
- Kartenelement
- Layerbaum
- Legende
- Übersichtskarte
- Navigations-Werkzeugkasten
- Infoabfrage
- Koordinatenanzeige
- Copyright
- Linien/Flächen-Messung
- Maßstabsauswahl
- Maßstabsleiste
- Spatial Reference System-Auswahl
- GPS-Position
- Druck
- WMS zur Anwendung hinzufügen
- Dokumentation unter http://doc.mapbender.org

CHAPTER

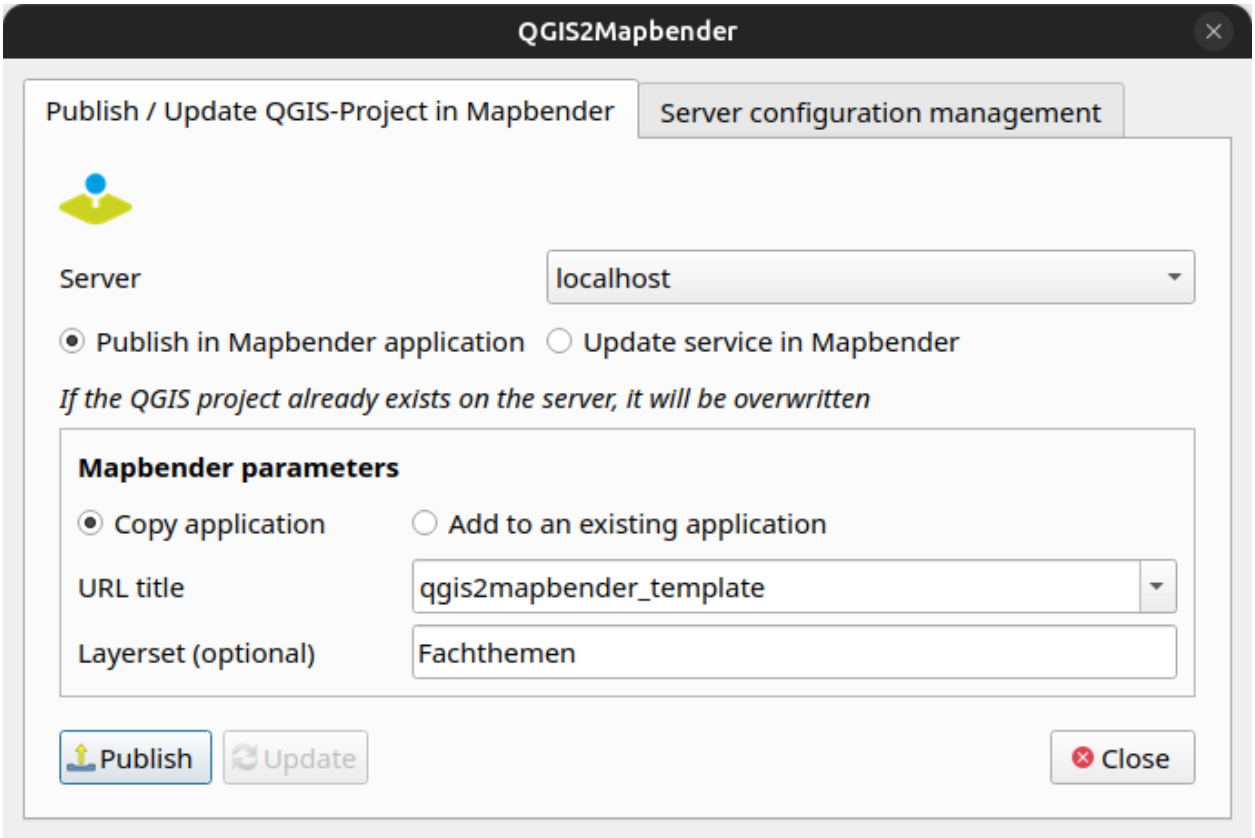
TEN

QGIS2MAPBENDER PLUGIN

10.1 Beschreibung

QGIS2Mapbender ist ein QGIS-Plugin, mit dem ein QGIS Server Projekt auf einen Server übertragen werden kann und anschließend der QGIS Server WMS in Mapbender veröffentlicht werden kann.

Sie finden den QGIS2Mapbender im QGIS Python Plugins Repository: https://plugins.qgis.org/plugins/qgis2mapbender



10.2 Installation und Anforderungen

Bitte beachten Sie, dass QGIS2Mapbender Version >= 1.0.0 eine Mapbender Version >= 4.1.2 benötigt.

10.2.1 Installation des Plugins

QGIS2Mapbender ist im QGIS Plugin Repository verfügbar. Die Installation ist direkt aus dem QGIS Plugin Repository über den QGIS Plugin Manager möglich. Klicken Sie dazu auf den Menüpunkt Plugins Verwalten und Installieren von Plugins. Alternativ kann eine Version auch hier heruntergeladen werden. Der gezippte Ordner kann manuell installiert werden. Klicken Sie auf den Menüpunkt Plugins Plugins verwalten und installieren. Wählen Sie im Dialog Plugin-Manager die Option “nicht installiert” und laden Sie das Zip hoch.

10.2.2 Anforderungen an Ihr lokales System

- Das QGIS-Projekt muss im selben Ordner wie die Daten gespeichert sein. Bitte beachten Sie, dass zusammen mit dem QGIS-Projekt auch alle Dateien in dem Ordner, der das QGIS-Projekt enthält, auf den Server hochgeladen werden.

10.2.3 Anforderungen an Ihren Server

- QGIS Server ist auf Ihrem Server installiert.
- Mapbender ist auf Ihrem Server installiert und konfiguriert.

10.2.4 Voraussetzungen für Ihre Mapbender-Installation

Apache

- Konfigurieren Sie die Apache-Autorisierung und das Mapbender-Upload-Verzeichnis api_upload_dir (siehe https://doc.mapbender.org/en/customization/api.html)

PHP

- Konfigurieren Sie die folgenden Parameter in der php.ini so, dass sie den Eigenschaften der Projekte entsprechen, die Sie auf den Server hochladen möchten. Denken Sie daran, dass der Ordner, der Ihr Projekt und Ihre Daten enthält, beim Hochladen auf den Server gezippt wird.
  - upload_max_filesize - die maximale Größe einer hochgeladenen Datei.
  - post_max_size - maximale Größe aller Daten, die über eine POST-Anfrage gesendet werden. Der Wert sollte gleich oder größer als upload_max_filesize sein.
  - max_execution_tine - hier wird die maximale Zeit in Sekunden festgelegt, die ein Skript für die Analyse der Eingabedaten benötigt.

Mapbender

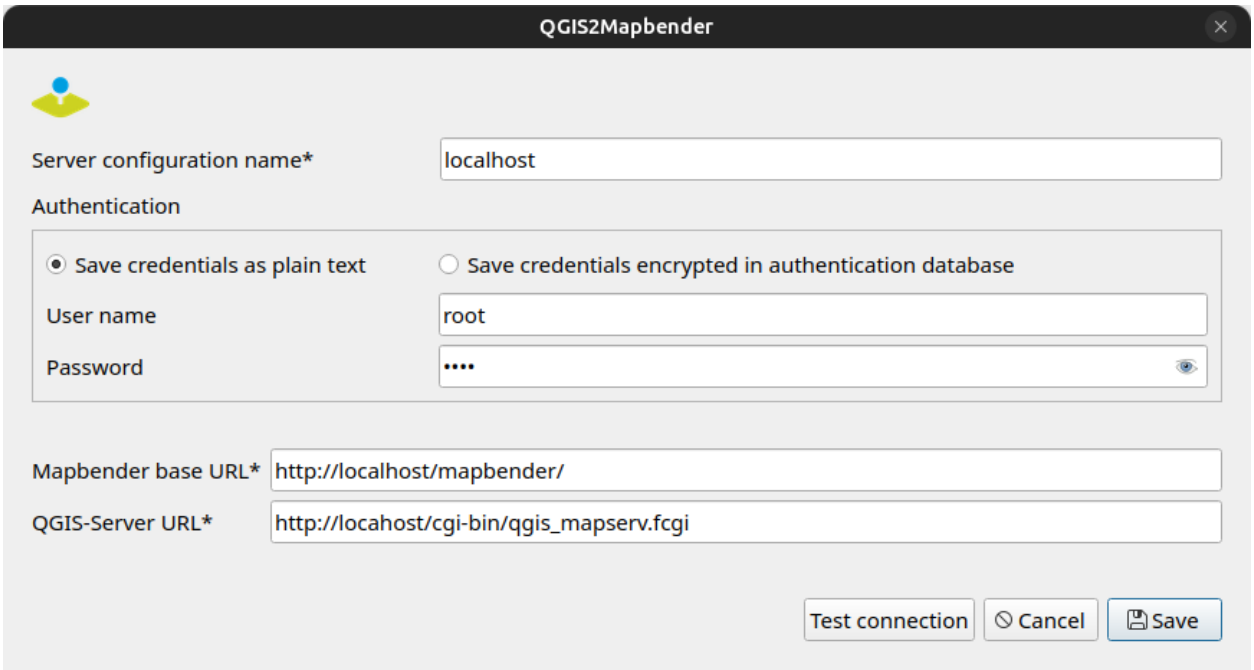
- Anwendung: Erstellen Sie mindestens eine Vorlage-Anwendung in Mapbender (die kopiert und zur Veröffentlichung eines neuen WMS verwendet werden kann) oder eine Anwendung, die direkt zur Veröffentlichung eines neuen WMS verwendet wird.
- Die Anwendungen sollten mindestens eine Instanz einer Karte und eines Layersets enthalten.

Hinweis: Das Feld „layerset“ in QGIS2Mapbender ist die ID oder der Name des zu verwendenden Layersets. Die Voreinstellung ist „main“ oder das erste Layerset in der Anwendung.

- Benutzer/Gruppen: Alle Mapbender-Benutzer, die berechtigt sein sollen, QGIS2Mapbender zu benutzen, benötigen spezielle Rechte. Es gibt nur eine Ausnahme und das ist der Mapbender-Superuser mit der ID 1, bei dem diese Berechtigung automatisch vergeben wird.
  - Der Benutzer/die Gruppe benötigt die globale Berechtigung access_api und upload_files, um alle Operationen auf der API durchführen und Dateien hochladen zu können.
  - Der Benutzer/die Gruppe benötigt die globale Berechtigung view_sources.
  - Der Benutzer/die Gruppe benötigt das globale Recht create_applications, um eine Anwendung zu kopieren.
  - Benutzer/Gruppe muss das Recht view_applications haben, um eine Anwendung zu kopieren.
  - Benutzer/Gruppe benötigt das globale Recht edit_applications, um eine Anwendung mit einer neuen Quelle zu aktualisieren.
  - Benutzer/Gruppe benötigt das globale Recht edit_soruces, um eine neue Quelle zu erstellen (veröffentlichen).
  - Der Benutzer/die Gruppe benötigt das globale Recht update_soruces, um eine Quelle neu zu laden.

10.2.5 Konfigurieren der Verbindung zum Server

Die folgende Abbildung zeigt eine typische Konfiguration der Verbindung zum Server.



Einige Anmerkungen zur Standardkonfiguration:

Parameter	Beschreibung	Beispiel
Mapbender Basis-URL	Link zu Ihrer Mapbender-Landingpage (Anwendungsübersicht)	http://localhost/mapbender/
QGIS Server URL	URL für den QGIS-Server Zugriff	http://localhost/cgi-bin/qgis_mapserv.fcgi

10.2.6 Docker

- QGIS Server und Mapbender können als Docker Container betrieben werden. Bitte stellen Sie sicher, dass das Mapbender Upload-Verzeichnis api_upload_dir den gleichen Pfad wie das QGIS Server Projektverzeichnis hat, da es im QGIS Server Request als Pfad im MAP Parameter verwendet wird.
- Ein Standard QGIS Projekt (Umgebung: QGIS_PROJECT_FILE) sollte nicht angegeben werden.

10.3 Unterstützung

info@wheregroup.com

10.4 Lizenz

Das Plugin ist unter der beigefügten GNU General Public License lizenziert.