

GeoStyler – Generic Styler for Geodata

Johannes Weskamm, terrestris GmbH & Co. KG

Jan Suleiman, terrestris GmbH & Co. KG

Daniel Koch, terrestris GmbH & Co. KG

Kai Volland, terrestris GmbH & Co. KG

Marc Jansen, terrestris GmbH & Co. KG

Christian Mayer, meggsimum

Johannes Weskamm



- Diploma in Geography
- Full-Stack Developer
@terrestris
Bonn, Germany
- GeoStyler Core Developer

✉ weskamm@terrestris.de

🐙 [@weskamm](https://github.com/weskamm)

Contents

1. What is GeoStyler?
2. How to use it?
 - The demo website
 - Code integration
 - On the commandline
 - The REST API
3. Latest features
4. What's next?
5. Getting help and information



What is GeoStyler?

- A set of tools that help creating and converting between different style formats
- Open Source, based on TypeScript
- Official OSGeo Community Project



See osgeo.org



What is GeoStyler?

Currently supported style formats:

- SLD (1.0 and 1.1)
- Mapfiles for MapServer
- QGIS
- CQL (Filters)
- OpenLayers
- MapBox
- GeoCSS




What is GeoStyler?


- Graphical editor for styling geodata
- Converter of styling formats



What is GeoStyler?



































Graphical editor for styling geodata




Docs 



Language: EN DE ES 中文 Compact: ☒ Symbolizer Renderer: OpenLayers SLD Load Style: Load Data: WFS Data Parser Examples


Name: Demo Style Classification

<input type="checkbox"/>		Name	Filter	Min. Scale	Max. Scale	Σ	
<input type="checkbox"/>		Hamburg	GEN = 'Hamburg'			1	0
<input type="checkbox"/>		Niedersachsen	GEN = 'Niedersachsen'			1	0
<input type="checkbox"/>		Bremen	GEN = 'Bremen'			1	0
<input type="checkbox"/>		Nordrhein-Westfalen	GEN = 'Nordrhein-Westfalen'			1	0
<input type="checkbox"/>		Hessen	GEN = 'Hessen'			1	0
<input type="checkbox"/>		Rheinland-Pfalz	GEN = 'Rheinland-Pfalz'			1	0
<input type="checkbox"/>		Baden-Württemberg	GEN = 'Baden-Württemberg'			1	0
<input type="checkbox"/>		Bayern	GEN = 'Bayern'			1	0
<input type="checkbox"/>		Saarland	GEN = 'Saarland'			1	0
<input type="checkbox"/>		Berlin	GEN = 'Berlin'			1	0
<input type="checkbox"/>		Brandenburg	GEN = 'Brandenburg'			1	0
<input type="checkbox"/>		Mecklenburg-Vorpommern	GEN = 'Mecklenburg-Vorpommern'			1	0
<input type="checkbox"/>		Sachsen	GEN = 'Sachsen'			1	0
<input type="checkbox"/>		Sachsen-Anhalt	GEN = 'Sachsen-Anhalt'			1	0
<input type="checkbox"/>		Thüringen	GEN = 'Thüringen'			1	0
<input type="checkbox"/>		Schleswig-Holstein	GEN = 'Schleswig-Holstein'			1	0

+Add Rule  Clone Rules  Remove Rules  Multi edit

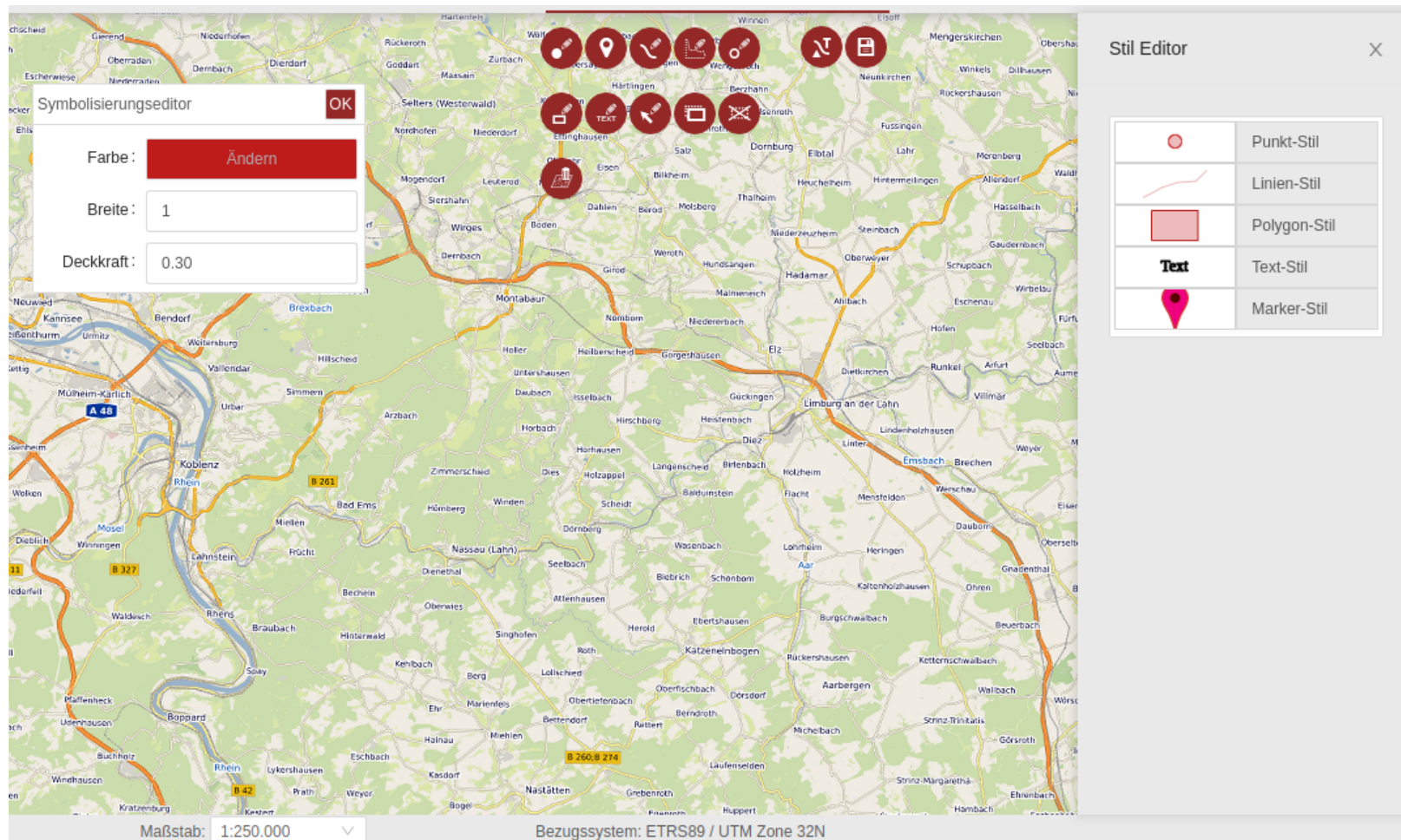
> Code Editor

> Preview Map



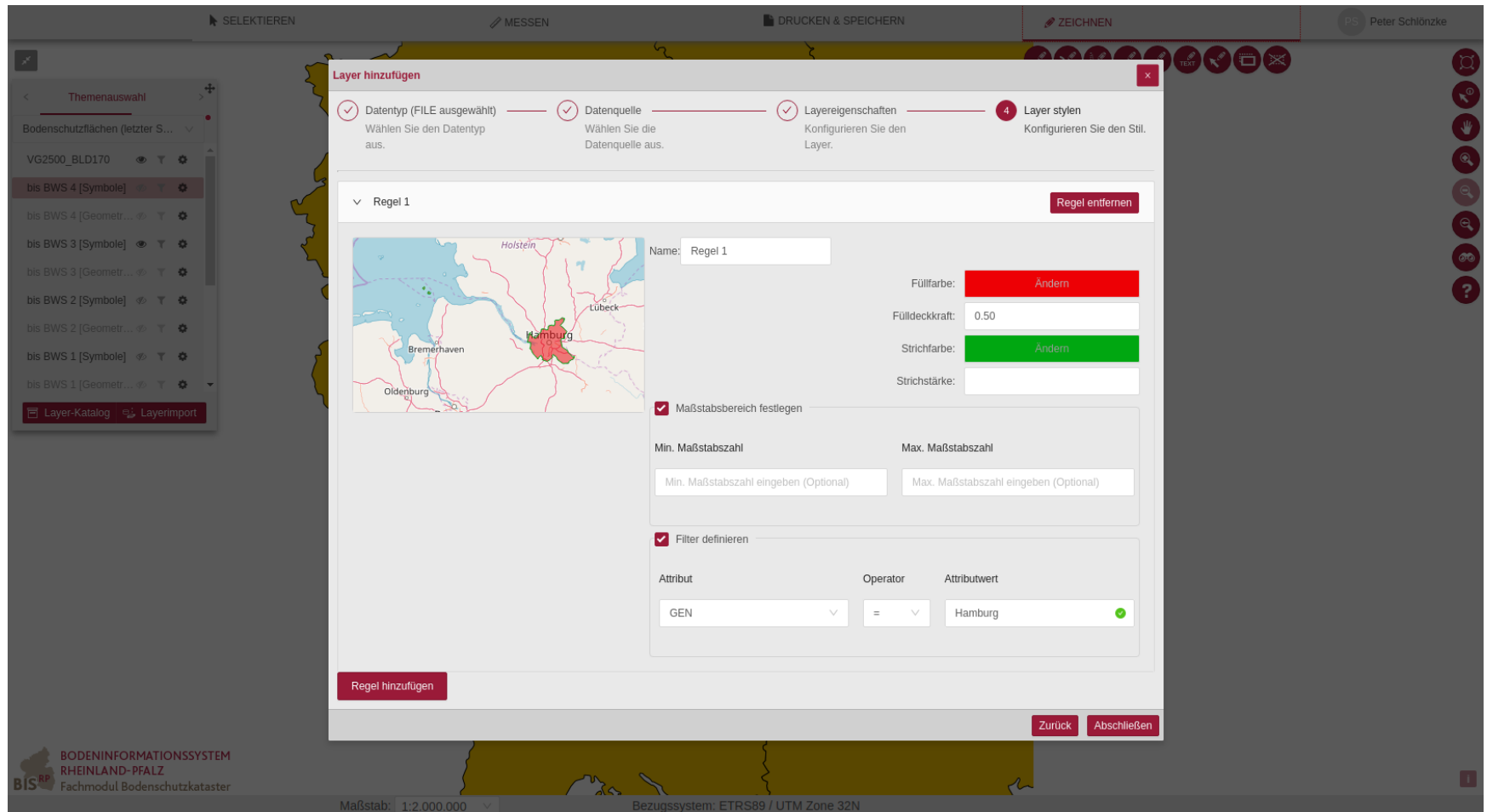
What is GeoStyler?

Graphical editor for styling geodata



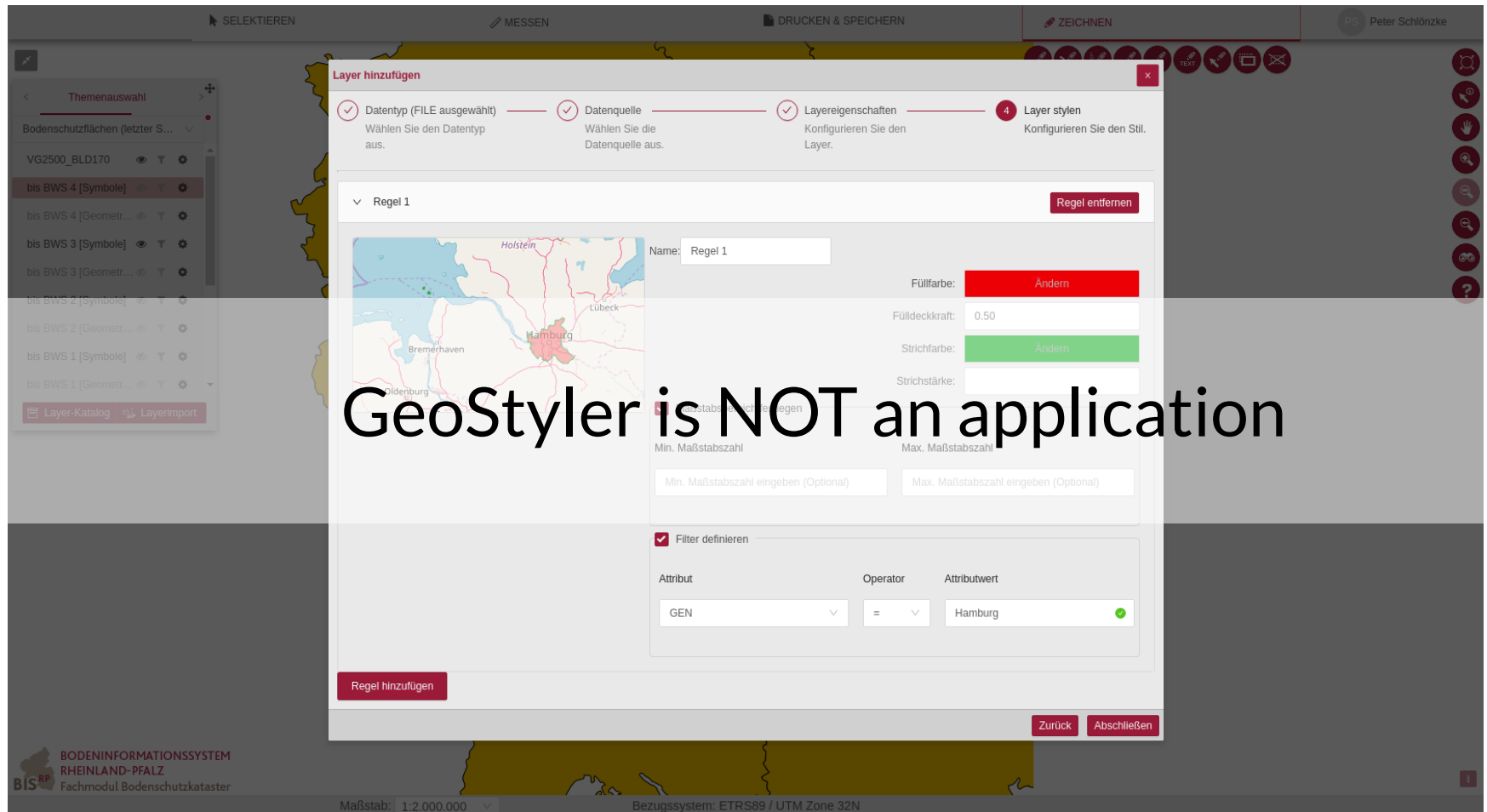
What is GeoStyler?

Graphical editor for styling geodata



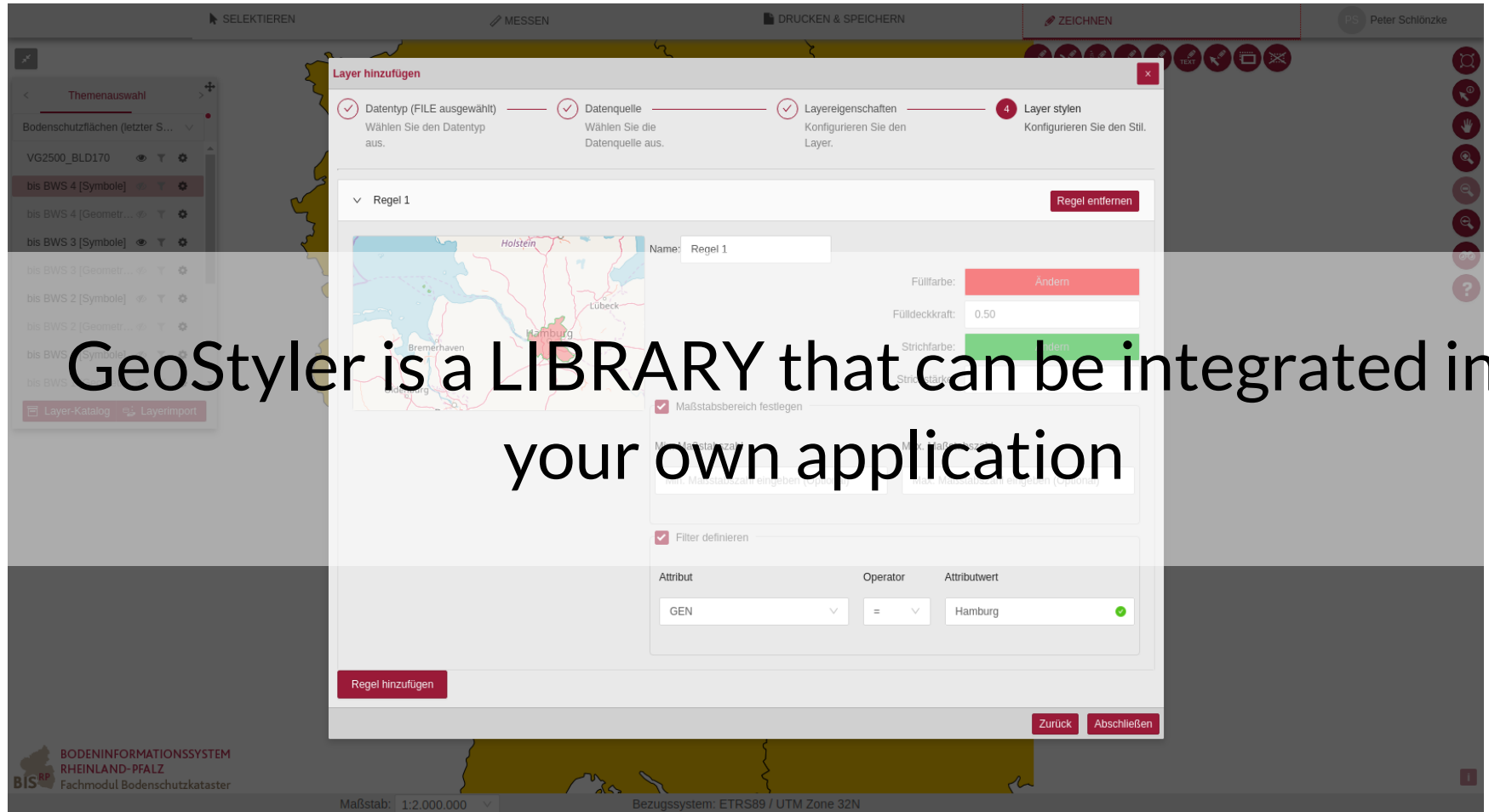
What is GeoStyler?

Graphical editor for styling geodata



What is GeoStyler?

Graphical editor for styling geodata

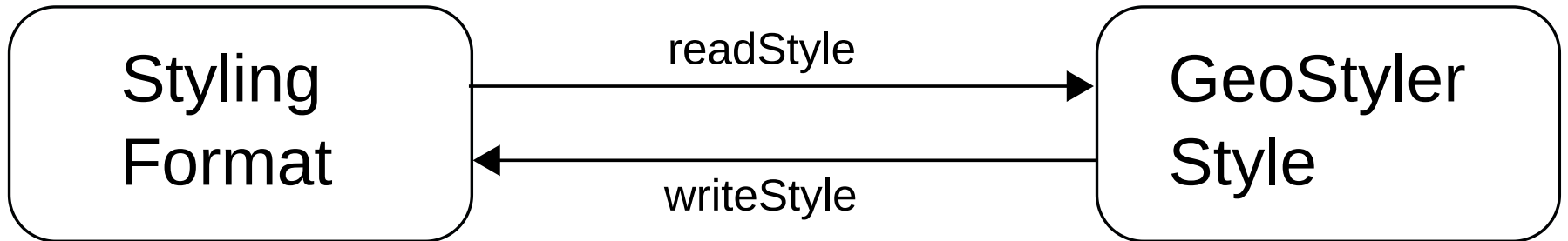


GeoStyler is a LIBRARY that can be integrated into your own application



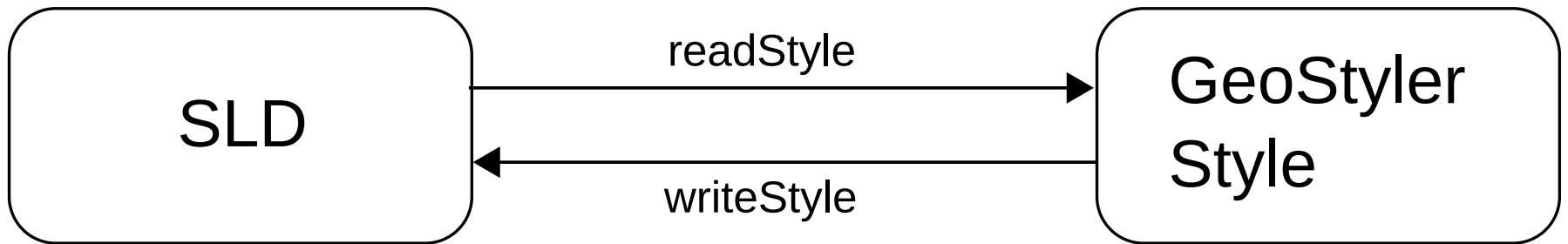
What is GeoStyler?

Converter of styling formats



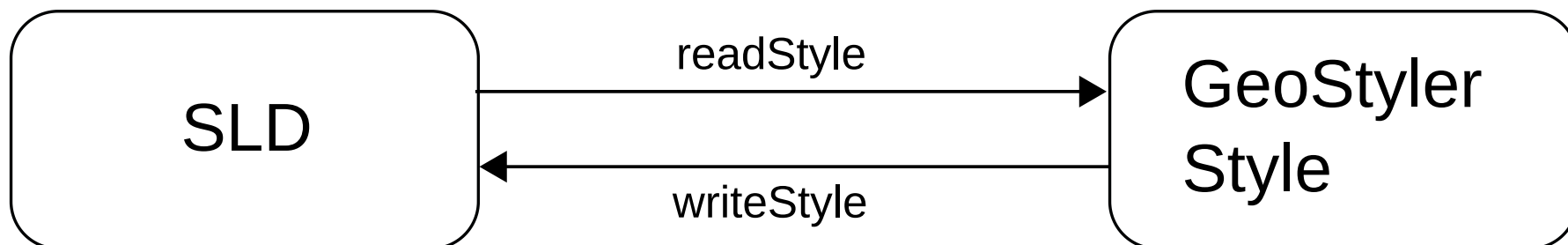
What is GeoStyler?

Converter of styling formats



What is GeoStyler?

Converter of styling formats



```
<!--?xml version="1.0" encoding="UTF-8" standalone="yes"?-->
<styledlayerdescriptor version="1.0.0" xsi:schemaLocation="http://www.opengis.net/sld StyledLayerDescriptor.xsd">
  <namedlayer>
    <name>Demo Style</name>
    <userstyle>
      <name>Demo Style</name>
      <title>Demo Style</title>
      <featuretypestyle>
        <rule>
          <name>Rule 1</name>
          <pointsymbolizer>
            <graphic>
              <mark>
```

```
{
  "name": "Demo Style",
  "rules": [
    {
      "name": "Rule 1",
      "symbolizers": [
        {
          "kind": "Mark",
          "wellKnownName": "circle"
        }
      ]
    }
  ]
}
```



What is GeoStyler?

Converter of styling formats



What is GeoStyler?

Converter of styling formats



What is GeoStyler?

Converter of styling formats



```
LAYER
  NAME "point_simple_point"
  TYPE POINT
  DATA "../shapes/points.shp"
  EXTENT -180 -90 180 90
  METADATA
    "wms_title" "point_simple_point"
    "wms_srs" "EPSG:4326"
  END
  CLASS
    STYLE
      SYMBOL "circle"
      COLOR 0 255 0
```

```
<!--?xml version="1.0" encoding="UTF-8" standalone="yes"?-->
<styledlayerdescriptor version="1.0.0" xsi:schemaLocation=
  <namedlayer>
    <name>Demo Style</name>
    <userstyle>
      <name>Demo Style</name>
      <title>Demo Style</title>
      <featuretypestyle>
        <rule>
          <name>Rule 1</name>
          <pointsymbolizer>
            <graphic>
              <mark>
```



How to use it?

Several ways how you can use GeoStyler:


- The demo website (limited, just a showcase!)
- Code integration
- On the commandline
- The REST API




How to use it?

The demo website

<https://geostyler.github.io/geostyler-demo/>

Docs 





































Language: EN DE ES 中文 Compact: ☐ Symbolizer Renderer: OpenLayers SLD Load Style: Load Data: WFS Data Parser Examples

Graphical Editor

Name:


Classification

<input type="checkbox"/>		Name	Filter	Min. Scale	Max. Scale	Σ	
<input type="checkbox"/>		Hamburg	GEN = 'Hamburg'			1	0
<input type="checkbox"/>		Niedersachsen	GEN = 'Niedersachsen'			1	0
<input type="checkbox"/>		Bremen	GEN = 'Bremen'			1	0
<input type="checkbox"/>		Nordrhein-Westfalen	GEN = 'Nordrhein-Westfalen'			1	0
<input type="checkbox"/>		Hessen	GEN = 'Hessen'			1	0
<input type="checkbox"/>		Rheinland-Pfalz	GEN = 'Rheinland-Pfalz'			1	0
<input type="checkbox"/>		Baden-Württemberg	GEN = 'Baden-Württemberg'			1	0
<input type="checkbox"/>		Bayern	GEN = 'Bayern'			1	0
<input type="checkbox"/>		Saarland	GEN = 'Saarland'			1	0
<input type="checkbox"/>		Berlin	GEN = 'Berlin'			1	0
<input type="checkbox"/>		Brandenburg	GEN = 'Brandenburg'			1	0
<input type="checkbox"/>		Mecklenburg-Vorpommern	GEN = 'Mecklenburg-Vorpommern'			1	0
<input type="checkbox"/>		Sachsen	GEN = 'Sachsen'			1	0
<input type="checkbox"/>		Sachsen-Anhalt	GEN = 'Sachsen-Anhalt'			1	0
<input type="checkbox"/>		Thüringen	GEN = 'Thüringen'			1	0
<input type="checkbox"/>		Schleswig-Holstein	GEN = 'Schleswig-Holstein'			1	0

+ Add Rule Clone Rules Remove Rules Multi edit

> Code Editor

> Preview Map



How to use it?

Code

<https://github.com/geostyler>

The screenshot shows the GitHub profile for 'geostyler'. The profile header includes a logo (a stylized globe with horizontal stripes) and the name 'geostyler' with a link to 'http://geostyler.org'. Below the header, there are tabs for 'Overview', 'Repositories' (23), 'Projects', 'Packages', 'Teams' (3), and 'People' (16). The 'Overview' tab is selected, showing a 'Pinned' section with six repositories: 'geostyler' (Generic Styler for geodata, TypeScript, 149 stars, 39 forks), 'geostyler-style' (Style declaration files for the GeoStyler project, TypeScript, 14 stars, 13 forks), 'geostyler-sld-parser' (GeoStyler-Style-Parser implementation for SLD, TypeScript, 31 stars, 22 forks), 'geostyler-openlayers-parser' (GeoStyler Style Parser implementation for OpenLayers styles, TypeScript, 24 stars, 21 forks), 'geostyler-demo' (GeoStyler Demo App, TypeScript, 7 stars, 11 forks), and 'geostyler-data' (Data declaration files for the GeoStyler project, JavaScript, 4 stars, 5 forks). To the right of the pinned repositories, there is a 'View as: Public' dropdown and a note about viewing the README. Below the pinned repositories, there is a 'People' section showing a grid of avatars, a 'Top languages' section showing TypeScript, JavaScript, and Scheme, and a 'Most used topics' section showing geostyler, openlayers, react, and sld. At the bottom, there is a 'Repositories' section with a search bar and a list of repositories, including 'geostyler-rest' (JavaScript, 2 stars, 5 forks, 1 commit, 5 issues, updated 2 days ago) and 'geostyler-sld-parser' (GeoStyler-Style-Parser implementation for SLD).

How to use it?

On the commandline

GeoStyler CLI

Conversion between styling formats:

```
geostyler -s qgis -t sld -o output.sld input.qml
```

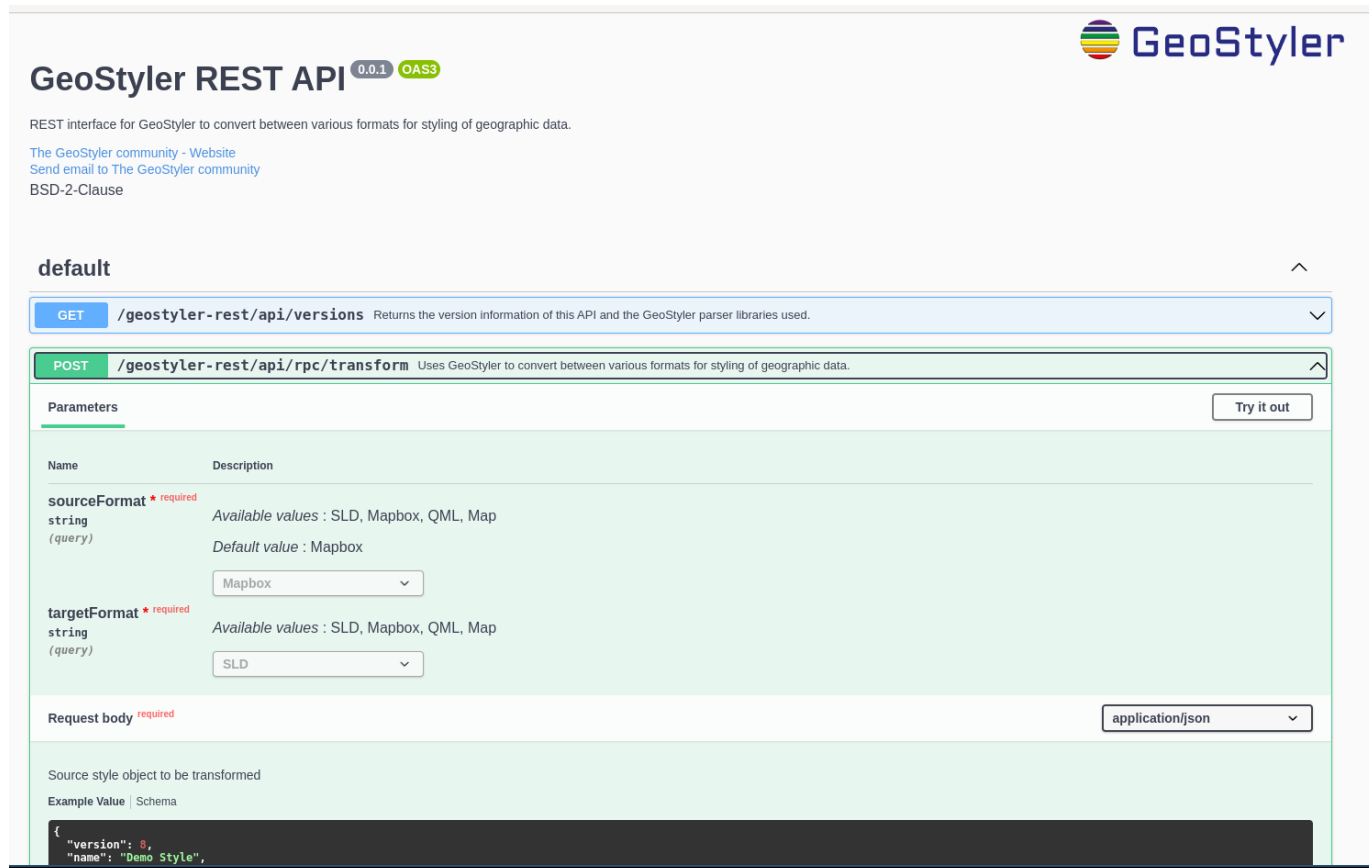
See [geostyler-cli](#)



How to use it?

The REST API

See [geostyler-rest](#)



The screenshot displays the GeoStyler REST API web interface. At the top right is the GeoStyler logo. The main heading is "GeoStyler REST API" with version "0.0.1" and "OAS3" tags. Below this is a description: "REST interface for GeoStyler to convert between various formats for styling of geographic data." and links to the community website, email, and license. The interface shows a "default" section with two API endpoints: a GET endpoint for "/geostyler-rest/api/versions" and a POST endpoint for "/geostyler-rest/api/rpc/transform". The POST endpoint is expanded, showing parameters for "sourceFormat" and "targetFormat", both required string query parameters. "sourceFormat" has available values "SLD", "Mapbox", "QML", and "Map", with a default value of "Mapbox". "targetFormat" has available values "SLD", "Mapbox", "QML", and "Map", with a default value of "SLD". Below the parameters is a "Request body" section, which is required and set to "application/json". It shows a "Source style object to be transformed" with an "Example Value" and "Schema" tab. The example value is a JSON object: {"version": 8, "name": "Demo Style", ...}. A "Try it out" button is located next to the parameters section.

GeoStyler REST API ^{0.0.1} OAS3

REST interface for GeoStyler to convert between various formats for styling of geographic data.

[The GeoStyler community - Website](#)
[Send email to The GeoStyler community](#)
[BSD-2-Clause](#)

default

GET /geostyler-rest/api/versions Returns the version information of this API and the GeoStyler parser libraries used.

POST /geostyler-rest/api/rpc/transform Uses GeoStyler to convert between various formats for styling of geographic data.

Parameters Try it out

Name	Description
sourceFormat * required string (query)	Available values : SLD, Mapbox, QML, Map Default value : Mapbox Mapbox
targetFormat * required string (query)	Available values : SLD, Mapbox, QML, Map SLD

Request body required application/json

Source style object to be transformed

Example Value Schema

```
{  
  "version": 8,  
  "name": "Demo Style",  
  ...  
}
```

Latest features

User manual



Latest features

Card layout

The screenshot displays the GeoStyler web application interface, highlighting the 'Card layout' feature. The interface is divided into two main panels: the 'Graphical Editor' on the left and the 'Code Editor' on the right.

Graphical Editor:

- Style:** A section for defining the style, including a 'Name' field (set to 'Demo Style') and a 'Title' field (placeholder: 'Enter Title').
- Rules:** A section for defining rules, featuring a 'Select' button and a list of rules. Two rules are visible:
 - No go area:** Represented by a blue square icon. Scale: 1:20000 — 1:300000. Summary: Σ · 16.
 - Parking:** Represented by a blue 'P' icon. Scale: 1: — 1:50000. Summary: Σ · 16.
- Buttons:** 'Add' and 'Classification' buttons are located at the bottom left.

Code Editor:

- Format:** A dropdown menu set to 'SLD Style Parser'.
- Code:** A text area displaying SLD XML code. The code defines a 'Demo Style' with a 'No go area' rule (blue square) and a 'Parking' rule (blue 'P' symbol).
- Buttons:** 'Copy to Clipboard' and 'Save as File' buttons are located at the bottom right.

Header:

- Language:** A dropdown menu with options: EN, DE, ES, FR, 中文.
- CardLayout (Beta):** A toggle switch.
- Symbolizer Renderer:** A dropdown menu with options: OpenLayers, SLD.
- Load Style:** A dropdown menu.
- Load Data:** A dropdown menu with options: WFS Data Parser, Examples.



Latest features

Enhanced GeoServer Plugin

GeoStyler

☐ Live preview enabled? (automatically saves the style)

Name: COUNTRIES

<input type="checkbox"/>		Name	Filter	
<input type="checkbox"/>		< 500.000	DENSITY < '500000'	
<input type="checkbox"/>		> 500.000	DENSITY > '500000'	
+ Add Rule Add Rule - Remove Rules Multi edit				

Font 12pt Height 300px

```
1 <?xml version="1.0" encoding="UTF-8" standalone="yes"?>
2 <StyledLayerDescriptor version="1.0.0" xsi:schemaLocation="http://www.opengis.net/sld StyledLayerDescriptor.xsd"
3   xmlns:ogc="http://www.opengis.net/ogc" xmlns:xlink="http://www.w3.org/1999/xlink" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
4   <NamedLayer>
5     <Name>COUNTRIES</Name>
6     <UserStyle>
7       <Name>COUNTRIES</Name>
8       <Title>COUNTRIES</Title>
9       <FeatureTypeStyle>
10        <Rule>
11          <Name>&lt; 500.000</Name>
12          <Filter xmlns="http://www.opengis.net/ogc">
13            <PropertyIsLessThan>
14              <PropertyName>DENSITY</PropertyName>
15              <Literal>500000</Literal>
16            </PropertyIsLessThan>
17          </Filter>
18          <PolygonSymbolizer>
19            <Fill>
20              <CssParameter name="fill">#40aa23</CssParameter>
21              <CssParameter name="fill-opacity">0.5</CssParameter>
22            </Fill>
23            <Stroke>
24              <CssParameter name="stroke">#3f4cc7</CssParameter>
25              <CssParameter name="stroke-width">2</CssParameter>
26            </Stroke>
27          </PolygonSymbolizer>
28        </Rule>
29      </FeatureTypeStyle>
30    </UserStyle>
31  </NamedLayer>
32</StyledLayerDescriptor>
```

Validate

Save

Apply

Cancel

Symbolizer Editor

Add

Kind: Fill

General

Fill-Color:  Change

Fill-Opacity: 1.00

Opacity:

Stroke-Opacity:

Outline-Color:  Change

Outline-Width:

Outline-Dasharray: + -

Graphic Fill

Classification

Latest features

Support for expressions

- Using functions in styles
 - Allows setting style properties attributively and feature-based
 - Allows for more compact and dynamic styles
 - Currently supported in GeoStyler style, SLD and CQL parser. More to come...

What's next?



What's next?

More styling formats

?? geostyler-arccgis-parser ??

on-going discussion...



What's next?

QGIS Plugin

Import/Export Styles in QGIS



What's next?

Next GeoStyler Code Sprint

Bugfixing, New Features, Community Building

When? TBA



Getting help and information

The Project Website

Infos, docs & tutorials on geostyler.org

See geostyler.org



Generic Styler for GeoData

Demo

Getting Started

Docs



Thank you for your attention



Imprint

Authors

Johannes Weskamm

terrestris GmbH & Co. KG

Jan Suleiman

Kölustr. 99, 53111 Bonn

Marc Jansen

Germany

Daniel Koch

Kai Volland

meggsimum

Christian Mayer

Schillerstraße 2a, 67112

Mutterstadt

Germany

License

Slides are published under [CC BY-SA](#) license.
terrestris.github.io/eurocarto2022/geostyler
PDF-Version, git repository