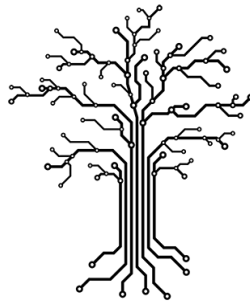


CLIMATETREE



Geoserver Developer Guide

Overview

Geoserver is used to deliver features to the frontend that then are displayed on the Map page. We are utilizing WFS with Parametric SQL views to allow for dynamic filtering. Our Geoserver layer queries our PostGIS database directly using a SQL query that pulls necessary information about all the places in our database

Goals and Audience:

Developers – who wish to learn how the Geoserver microservice is being implemented and want to scale it further

Tools, technologies, and servers used

Geoserver microservice utilizes the below tools and technologies:

1. Container - Docker
2. Database - PostgreSQL DB - PostGIS
3. Repository – Github

Running The Container Locally

1. Click the repository link below: <https://github.com/climatetree/geoserver-docker>
2. Clone the repository
3. Ensure you have Geoserver and Docker installed on your computer
4. Run `docker pull kartoza/geoserver`
5. Run `docker build -t .`
6. Run `docker run -it -p 8080:8080`

API documentation link

<https://documenter.getpostman.com/view/7834710/Szf6X8Wf?version=latest>

WFS

The Web Feature Service (WFS) is a standard created by the Open Geospatial Consortium (OGC) for creating, modifying and exchanging vector format geographic information on the Internet using HTTP. A WFS encodes and transfers information in Geography Markup Language (GML), a subset of XML.

Climate Tree is utilizing WFS version 2.0 which is compatible with previous versions 1.1.0 and 1.0.0.

Example API Call:

```
https://climatetree-api-gateway.azurewebsites.net/geoserver/wfs?
service=wfs&version=2.0.0&request=GetFeature&typeName=ClimateTree:Similar
_Places&outputFormat=application/json&viewparams=TYPE_ID_1:1;TYPE_ID_2:2;TYPE_ID_3:3;TY
PE_ID_4:4
```

Parameters:

The 'viewparams' parameter requires the value to be a semicolon delimited set of nested params that will then be used in the SQL query. The params in the set must have a ':' separating it from its value. All requests to this layer require **at least one TYPE_ID parameter. Otherwise, 0 places will be returned.**

TYPE_ID_1:1 — 'TYPE_ID_1' corresponds to places in our database categorized as 'State'. The value for this parameter should always be '1'.

TYPE_ID_2:2 — 'TYPE_ID_2' corresponds to places in our database categorized as 'Nation'. The value for this parameter should always be '2'.

TYPE_ID_3:3 — 'TYPE_ID_3' corresponds to places in our database categorized as 'County'. The value for this parameter should always be '3'.

TYPE_ID_4:4 — 'TYPE_ID_4' corresponds to places in our database categorized as 'Urban Extent'. The value for this parameter should always be '4'.

POPULATION_LOW — Type: Decimal. Value must be positive. This parameter is the lower range for filtering places by total population

POPULATION_HIGH — Type: Decimal. Value must be positive. This parameter is the upper range for filtering places by total population.

CARBON_LOW — Type: Decimal. This parameter is the lower range for filtering places by total carbon emissions in kg/yr.

CARBON_HIGH — Type: Decimal, This parameter is the upper range for filtering places by total carbon emissions in kg/yr.

POPULATION_DENSITY_LOW — Type: Decimal. Value must be positive. This parameter is the lower range for filtering places by population density as per square km.

POPULATION_DENSITY_HIGH — Type: Decimal. Value must be positive. This parameter is the upper range for filtering places by population density as per square km

PERCAP_CARBOIN_LOW — Type: Decimal. This parameter is the lower range for filtering places by carbon emissions per capita in kg/yr per square km.

PERCAP_CARBOIN_HIGH — Type: Decimal. This parameter is the upper range for filtering places by carbon emissions per capita in kg/yr per square km.

Max Limit:

The maximum number of results returned for any request to the Similar Places layer is 200.

Known issues and risks

- 1) WFS is slowest service that Geoserver offers for serving layer data
- 2) The Database schema impacts how quickly the SQL query runs and the layer is being served from Geoserver.

Future features:

- 1) Address the Database schema to allow for a faster performing SQL query for places
- 2) Add more layers and even geospatial analysis into Geoserver to make the Map super cool!