

Alternative Fuel Stations

A man in a light blue t-shirt and dark shorts is standing at a modern, illuminated electric vehicle charging station at night. He is holding a charging cable and appears to be plugging it into a car. The station is a tall, rectangular unit with a glowing blue light strip running vertically down its side. To the right, another similar charging station is visible, also illuminated with blue light. The background is dark, with some blurred lights from a building or street. The overall scene is futuristic and clean.

Nnamdi Ezeoke
Chris Heng
Beenish Mehboob
Sneha Thomas
Yi Zhang



Agenda

- ⚡ Overview
- ⚡ API Selection
- ⚡ Database Creation
- ⚡ Map Creation
- ⚡ Chart Creation
- ⚡ Flask Connection
- ⚡ Demonstrations

Overview

- Curiosity about the distribution of alternative fuel stations across the country and compare to the more popular EV stations
- We created flask-enabled, interactive maps and charts utilizing MongoDB that housed alternative fuel station data; obtained using an API.



Documentation

Use this Web service documentation to access and use energy data via application programming interfaces (APIs) in these renewable energy categories. You'll need a key to use these APIs. [Sign up for an API key.](#)

Buildings

Services related to energy efficiency and the use of renewable technologies in residential and commercial buildings.

Electricity

Services associated with electricity costs, generation, transmission, delivery, and monitoring.

Energy Optimization

Tools and models to optimize renewable energy, conventional generation, and energy storage systems for buildings, campuses, communities, and microgrids.

Partnering

A collection of APIs providing data related to energy technologies, experts, and patents associated with the Department of Energy's (DOE) Lab Partnering Service (LPS).

Solar

Access data and analysis services that provide access to solar resource data and NREL models.

Transportation

Use this Web service documentation to access data about alternative transportation technologies.

Wave

Access data and analysis services that provide access to wave resource data and NREL models.

API Selection

Database Creation

```
import requests
from pprint import pprint
from config import api_key
import json
from pymongo import MongoClient
import pandas as pd

/ 2.2s
```

Enter the following into git terminal to create and add to fuel station collection:

```
mongoimport --type json -d ev_db -c fuel_stations --drop --jsonArray ev_bd.json
```

Modify the text for each fuel type to add other documents to the existing fuel station collection:

```
mongoimport --type json -d ev_db -c fuel_stations --jsonArray ev_cng.json
```

```
urlelec = f"https://developer.nrel.gov/api/alt-fuel-stations/v1.json?api_key={api_key}&fuel_type=ELEC"
response = requests.get(urlelec)

elec = response.json()
elec_json = json.dumps(elec['fuel_stations'], indent=4, sort_keys=False)

with open("ev_elec.json", "w") as outfile:
    outfile.write(elec_json)
# print(elec_json)
```

```
urlbd = f"https://developer.nrel.gov/api/alt-fuel-stations/v1.json?api_key={api_key}&fuel_type=BD"
response = requests.get(urlbd)

bd = response.json()
bd_json = json.dumps(bd['fuel_stations'], indent=4, sort_keys=False)

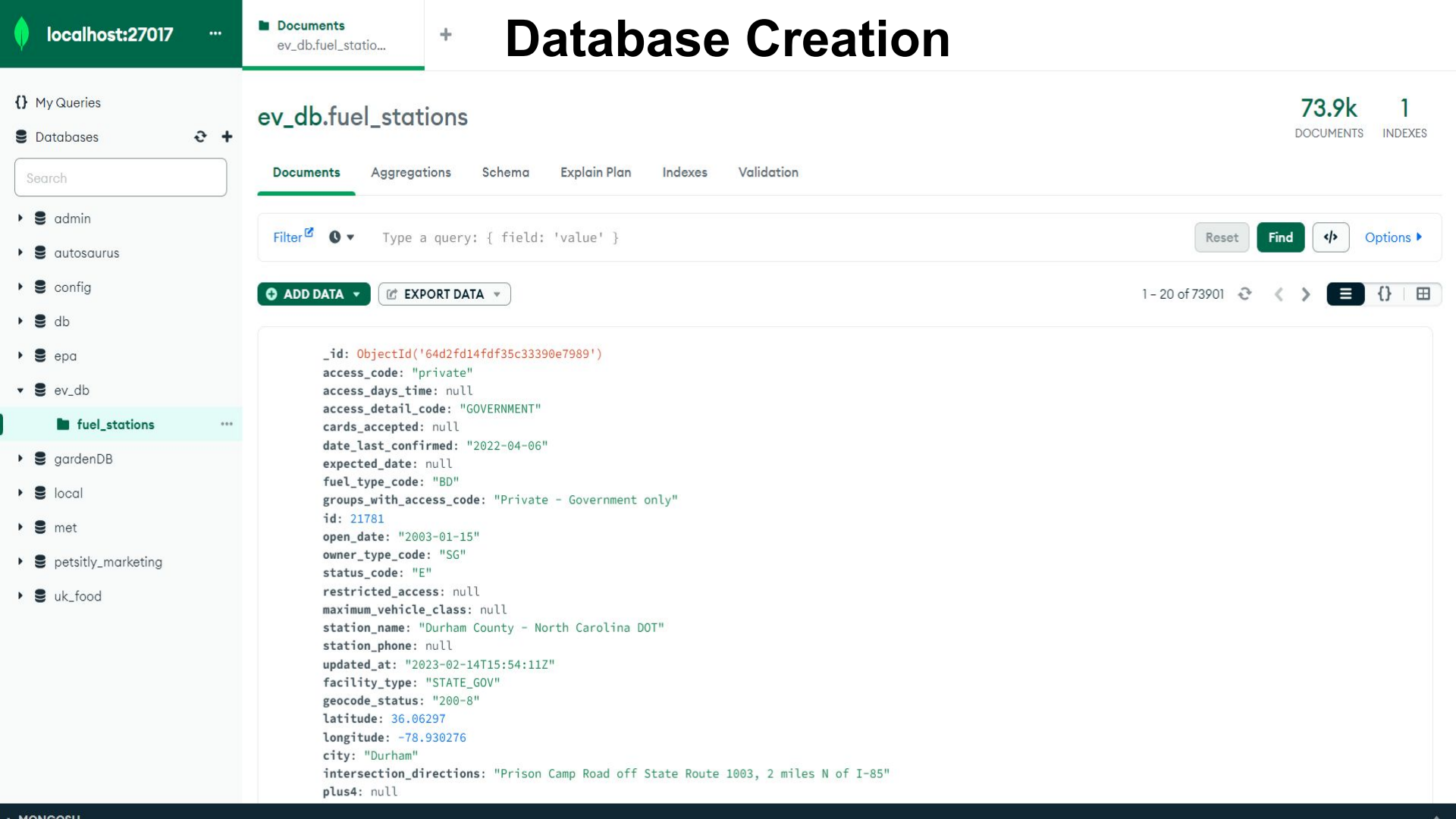
with open("ev_bd.json", "w") as outfile:
    outfile.write(bd_json)
# print(bd_json)
```

```
urle85 = f"https://developer.nrel.gov/api/alt-fuel-stations/v1.json?api_key={api_key}&fuel_type=E85"
response = requests.get(urle85)

e85 = response.json()
e85_json = json.dumps(e85['fuel_stations'], indent=4, sort_keys=False)

with open("ev_e85.json", "w") as outfile:
```

```
[
{
  "access_code": "private",
  "access_days_time": null,
  "access_detail_code": null,
  "cards_accepted": null,
  "date_last_confirmed": "2023-04-06",
  "expected_date": null,
  "fuel_type_code": "CNG",
  "groups_with_access_code": "Private",
  "id": 17,
  "open_date": "2010-12-01",
  "owner_type_code": "I",
  "status_code": "E",
  "restricted_access": null,
  "maximum_vehicle_class": "MD",
  "station_name": "Spire - Montgomery Operations Center",
  "station_phone": null,
  "updated_at": "2023-05-30T18:46:28Z",
  "facility_type": "STANDALONE_STATION",
  "geocode_status": "200-9",
  "latitude": 32.367916,
  "longitude": -86.267021,
  "city": "Montgomery",
  "intersection_directions": null,
  ...
  "groups_with_access_code_fr": "Priv\u00e9",
  "ev_pricing_fr": null
}
```



Mapping Using Leaflet

Leaflet



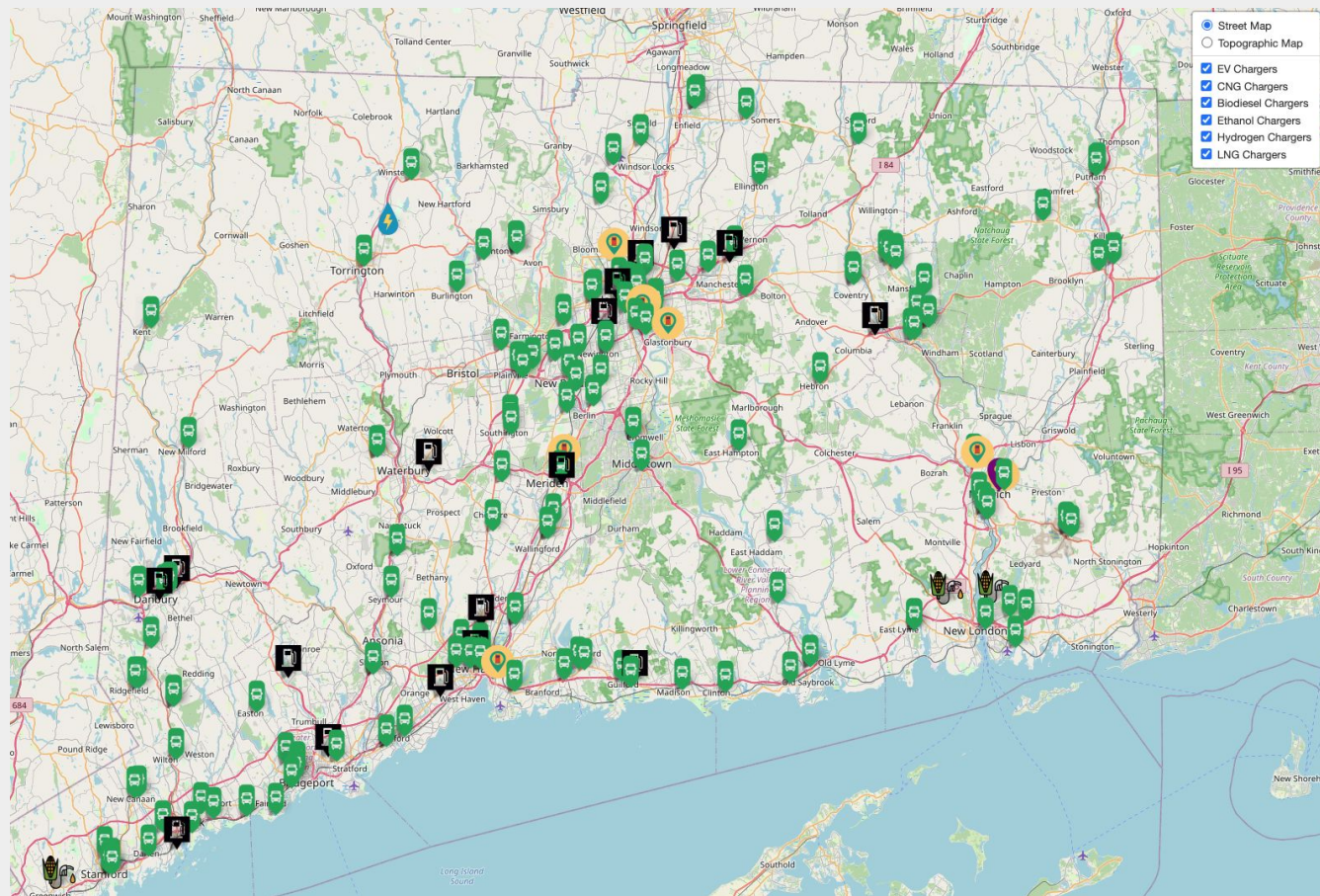
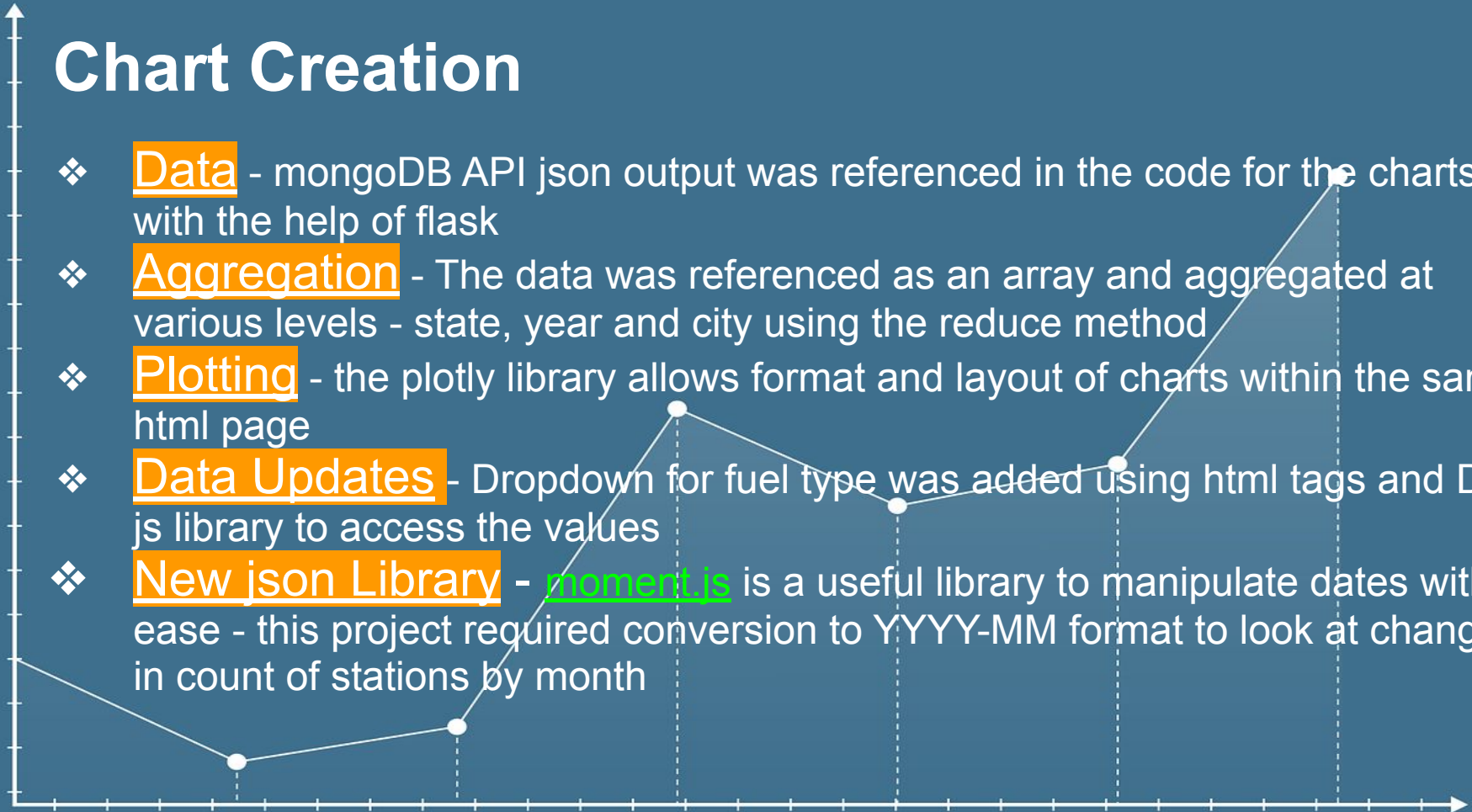


Chart Creation

- ❖ **Data** - mongoDB API json output was referenced in the code for the charts, with the help of flask
- ❖ **Aggregation** - The data was referenced as an array and aggregated at various levels - state, year and city using the reduce method
- ❖ **Plotting** - the plotly library allows format and layout of charts within the same html page
- ❖ **Data Updates** - Dropdown for fuel type was added using html tags and D3.js library to access the values
- ❖ **New json Library** - **moment.js** is a useful library to manipulate dates with ease - this project required conversion to YYYY-MM format to look at change in count of stations by month

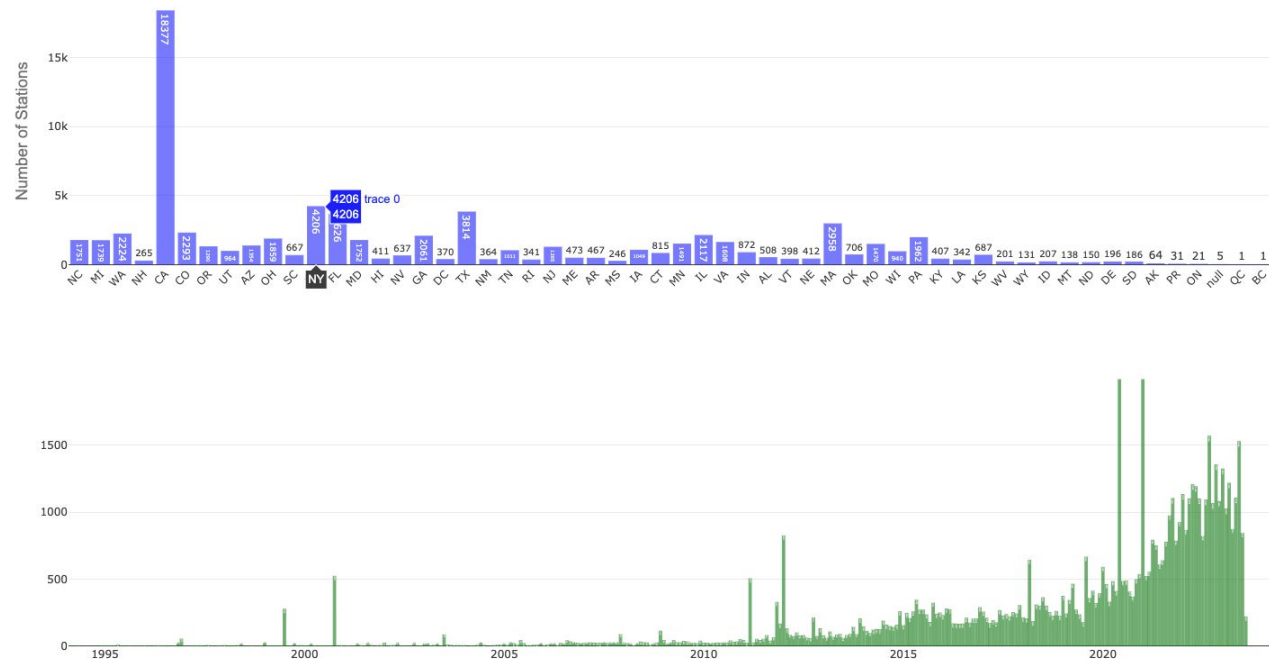


Alternative Fuel Stations Distribution

ALL Stations

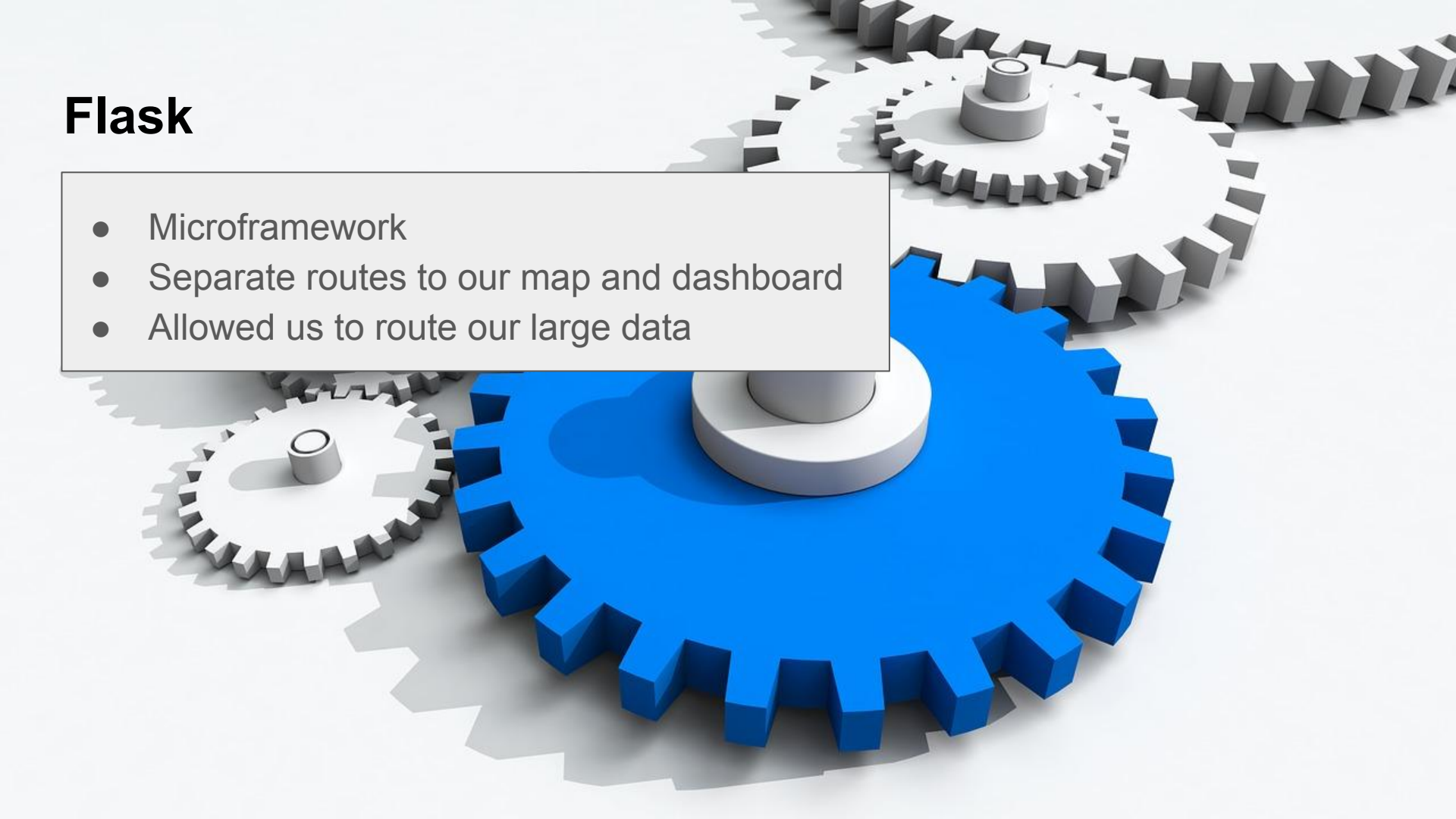


Number of Stations by Year and State



Flask

- Microframework
- Separate routes to our map and dashboard
- Allowed us to route our large data



App Demonstration

<http://127.0.0.1:50091/>



Reflection

- **Infrastructure for multiple alternative fuel sources still lacking:**
 - Putting all our eggs in one basket is a risky move for nation because if our energy grid cannot support the influx of new EV drivers then we need to be prepared to fuel other types of vehicles
- **Barriers when working with large datasets:**
 - Physical limitation of your machine
 - Being able to easily share it
 - Limitations on number of metrics that could be used for insights
 - Loading charts efficiently to reduce delays

Questions?

