

API: An alternative to www.eia.gov

- Why would someone **not** want to use eia.gov?
 - Personal preference: They don't like the site, our table browsers, or Tableau visualizations
 - The UX doesn't let them manipulate the data the way that they want
 - They need a data *quick hit*; they know exactly what they want and just need the raw numbers
 - They want to check our data with high frequency
 - Using *screen scrapers* to harvest site content is expensive both for user and EIA
 - PDF, CSV *exports* pre-bake data in inflexible formats

(Our Excel/Sheet add-ins call the API! Technically, those aren't exports.)
- For your use case: No web scraping! No clicking through data browsers!

An APIv1 call (semi-RESTful)

- A **target**: API host
- A **verb**: HTTP Method: GET, POST, etc
- A **series name**: Uniquely, IDs specific, data facets pre-baked
- Optional **modifiers** (row count, start and end)
- Data is pre-baked into date-time series

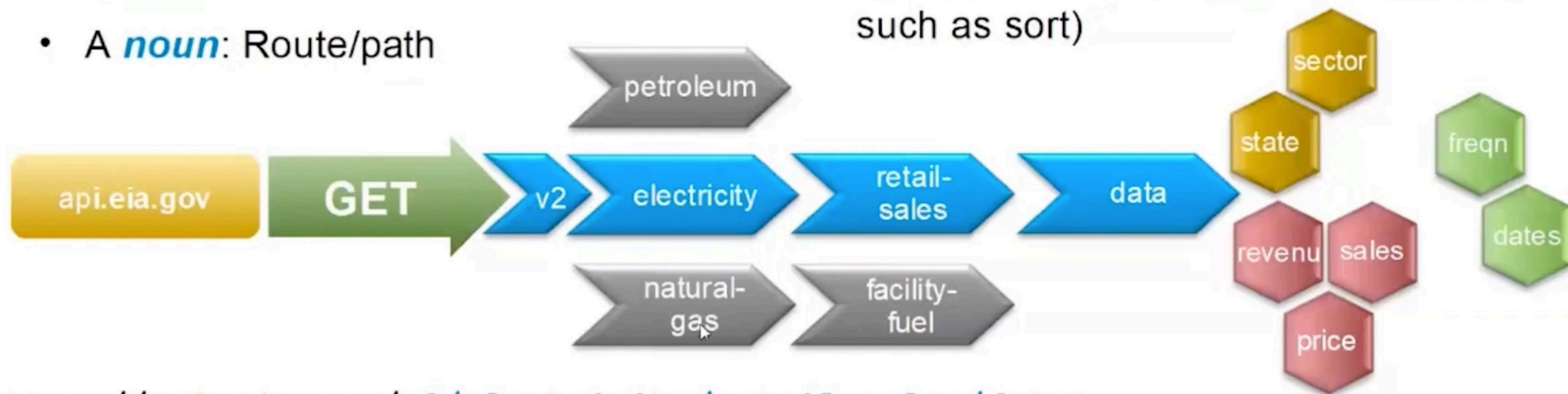


`https://api.eia.gov/series/?series_id=ELEC.SALES.CO-RES.M&api_key=xxxxxx`

APIv2: RESTful APIs and routes

Data can be discovered, selected, and interacted with

- A **target**: base URI, or API host
- A **verb**: HTTP Method: GET, POST, etc
- A **noun**: Route/path
- Optional **adverbs** (facets that specify specific data)
- Optional **modifiers** (parameters/filters such as sort)



```
https://api.eia.gov/v2/electricity/retail-sales/data?  
data[]=price&facets[sectorid][]=RES&facets[stateid]  
[]=CO&frequency=monthly&api_key=xxxxxx
```

Getting Setup

- Register for a key
- Use the table browser to get warmed up. It will build API calls for you:

<https://www.eia.gov/opendata/browser/>

- Read the documentation! 😊

<https://www.eia.gov/opendata/documentation.php>

- Now let's go hands on...

(I'll improv, but this is the same use case as in the documentation, so you can follow along)

Let's start at the top

(Remember, you'll use your own API key instead of this one)

`https://api.eia.gov/v2/?api_key=cbd67337cbc4bed426ef3682226bdab2`



API returns the routes available to you. This API is *describing itself*.

(You could also put requests in HTTP headers, but for this demo I'll put everything in the URL)

```
{
  "response": {
    "id": "",
    "name": "",
    "description": "",
    "routes": [
      {
        "id": "coal",
        "name": "Coal",
        "description": "EIA coal energy data"
      },
      {
        "id": "crude-oil-imports",
        "name": "Crude Oil Imports",
        "description": "Crude oil imports by country to destination, \r\n\r\n includes type, grade, quantity. Source: EIA-814 Interactive data \r\n\r\n product: www.eia.gov/petroleum/imports/companylevel/"
      },
      {
        "id": "electricity",
        "name": "Electricity",
        "description": "EIA electricity survey data"
      },
      {
        "id": "international",
        "name": "International",
        "description": "Country level production, consumption, imports, exports by energy source (petroleum, natural gas, electricity, renewable, etc.) \r\n\r\n Interactive product: https://www.eia.gov/international/data/world"
      },
      {
        "id": "natural-gas",
        "name": "Natural Gas",
        "description": "EIA natural gas survey data"
      },
      {
        "id": "nuclear-outages",

```

Drilling down

- Let's explore electricity.

`https://api.eia.gov/v2/electricity/?
api_key=cbd67337cbc4bed426ef3682226bdab2`



`https://api.eia.gov/v2/electricity/retail-sales/?
api_key=cbd67337cbc4bed426ef3682226bdab2`



- When we get to the last route, API shows the `data[]` object.

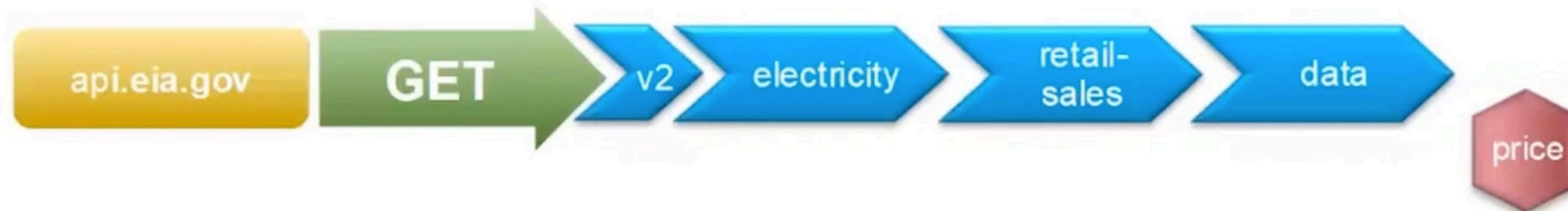
Specifying data

We must end at the `/data` node to get data:

```
https://api.eia.gov/v2/electricity/retail-sales/data  
/?api_key=cbd67337cbc4bed426ef3682226bdab2
```

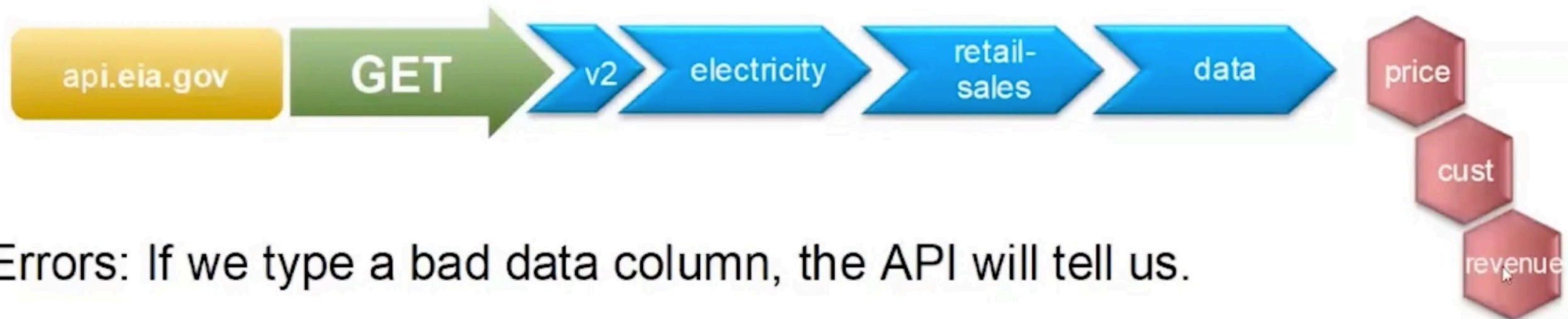
and specify which columns we want returned:

```
https://api.eia.gov/v2/electricity/retail-sales/data?data[]=price  
?api_key=cbd67337cbc4bed426ef3682226bdab2
```



Asking for more data columns

```
https://api.eia.gov/v2/electricity/retailsales/data ?  
data[]=price&data[]=revenue&data[]=customers  
&api_key=cbd67337cbc4bed426ef3682226bdab2
```



- Errors: If we type a bad data column, the API will tell us.
- Warnings: If we select too much data, the API will warn us, and return what it can.

Raw

Parsed

```
{
  "response": {
    "warnings": [
      {
        "warning": "incomplete return",
        "description": "The API can only return 5000 rows in JSON format. Please consider constraining your request with facet, start, or end, or using offset to paginate results."
      }
    ],
    "total": 96348,
    "dateFormat": "YYYY-MM",
    "frequency": "monthly",
    "data": [
      {
        "period": "2015-03",
        "stateid": "ME",
        "stateDescription": "Maine",
        "sectorid": "RES",
        "sectorName": "residential",
        "price": 14.81,
        "customers": 702745,
        "revenue": 63.11465,
        "price-units": "cents per kilowatthour",
        "customers-units": "number of customers",
        "revenue-units": "million dollars"
      }
    ]
  }
}
```