# User Guide

## Introduction

This website was built in order to host an industrial energy efficiency data explorer. Key components of the website include a dashboard, access to the underlying integrated industrial dataset and its creation, as well as documentation necessary for users to utilize and update as new data becomes available.

The website can be navigated through the buttons in the header, located along the top-left corner.

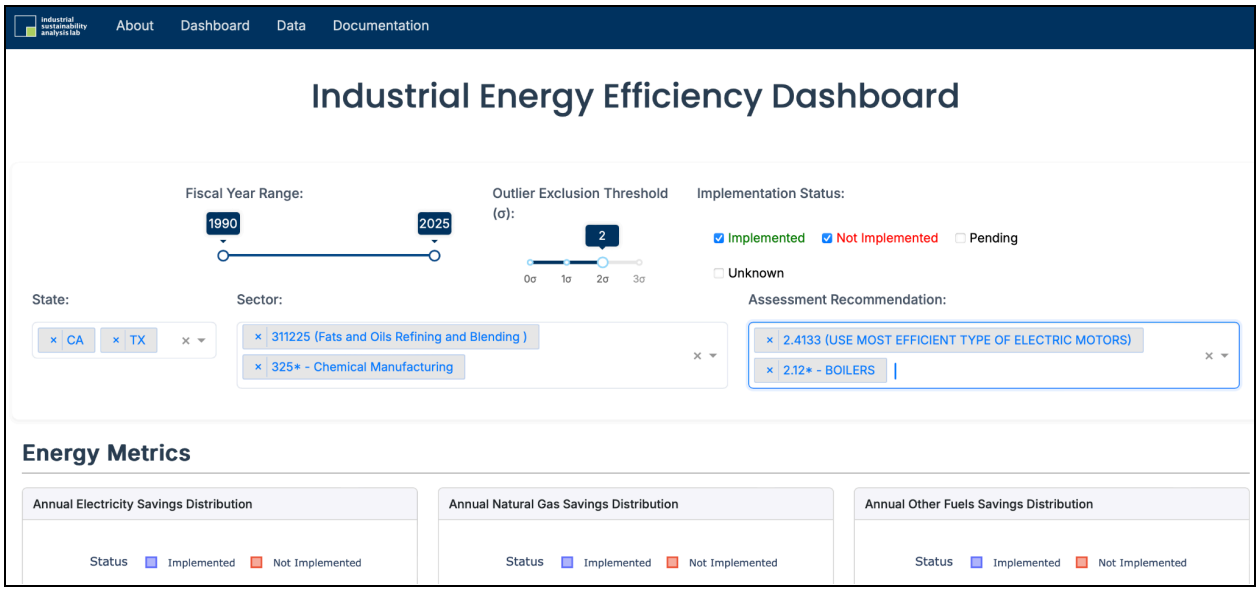

Clicking on the following buttons redirects to the following pages:

## About Page

The About page features information about the dashboard as well as the capstone team, data, and code behind the project. To view the summaries, click on the downward arrows (bordered in red below).


## Dashboard Page

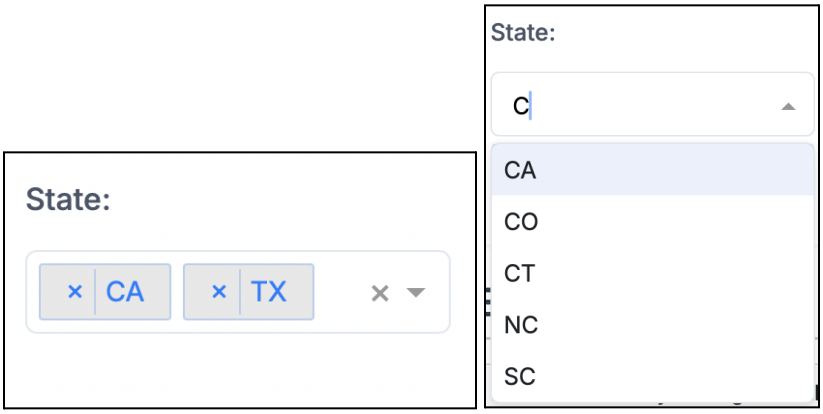
The dashboard presents a centralized interface for exploring industrial energy audit data through potential and realized energy, fiscal and emissions metrics, across facilities and recommendations. It is designed to support data exploration, policy analysis, and decarbonization strategies in industrial energy use.



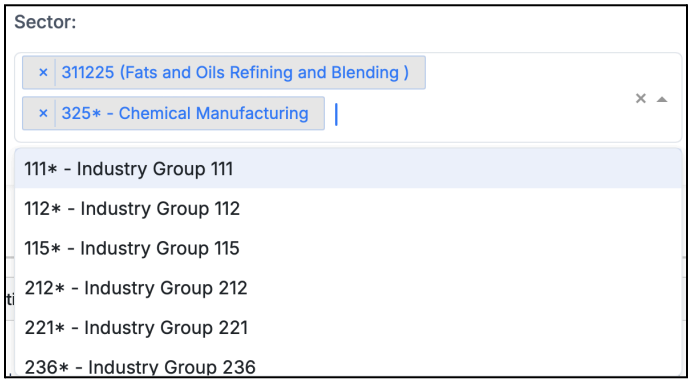

## Dashboard Filters

The dashboard includes multiple filters to allow users to tailor data visualizations around their research and modeling needs. A list of filters can be found below.

- State: Select any of the 50 US states, as well as Puerto Rico. Select one, multiple, or leave the selection empty to include all states. Scroll or type to find desired codes.

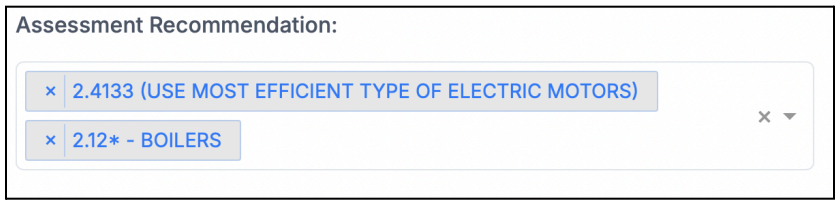



- Sector (NAICS Code): Select relevant NAICS codes or industries. Select one, multiple, or leave the selection empty to include all sectors. Scroll or type to find desired codes.

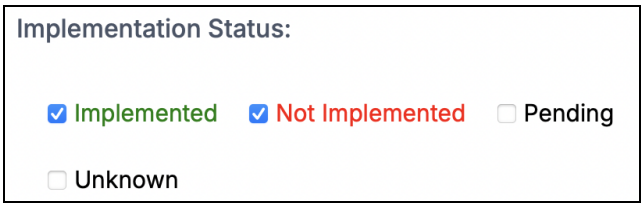




- Fiscal Year Range: Narrow down results to a specific time frame (within the 1990 - 2025 time range) via the sliding scale. Click and drag the white buttons to the desired year(s).

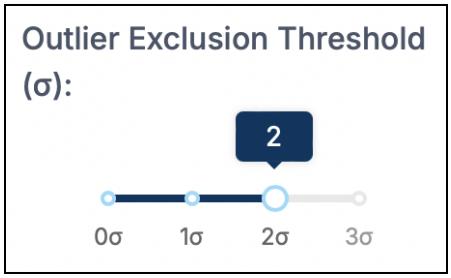

- Assessment Recommendation Code (ARC): Select types of energy-saving recommendations or overarching recommendation categories - produced from Industrial Assessment Centers' (IAC) audits. Select one, multiple, or leave the selection empty to include all recommendations. Scroll down or type to select desired codes.




- Implementation Status: Filter by whether an IAC recommendation was implemented, not implemented, pending, or unknown. Single, multiple, and none can be checked.




- Outlier Exclusion Threshold (σ): Filter the data to focus the visualized data on the desired distribution. Click and drag the white button to choose the desired visualized standard deviations from the mean.




**Visualizations**

## The Importance of Box Plots

A box plot (or box-and-whisker plot) summarizes the distribution of the selected inputs using five key values: the minimum, first quartile (Q1), median (Q2), third quartile (Q3), and maximum. The box spans from Q1 to Q3, representing the middle 50% of the data (interquartile range), with a line inside showing the median. Whiskers extend from the box to the smallest and largest values within 1.5 times the interquartile range, while any points beyond are considered outliers. The shape of the box and position of the median indicate the data's spread and skewness, helping identify symmetry, variability, and unusual values.

## Visualized Metrics

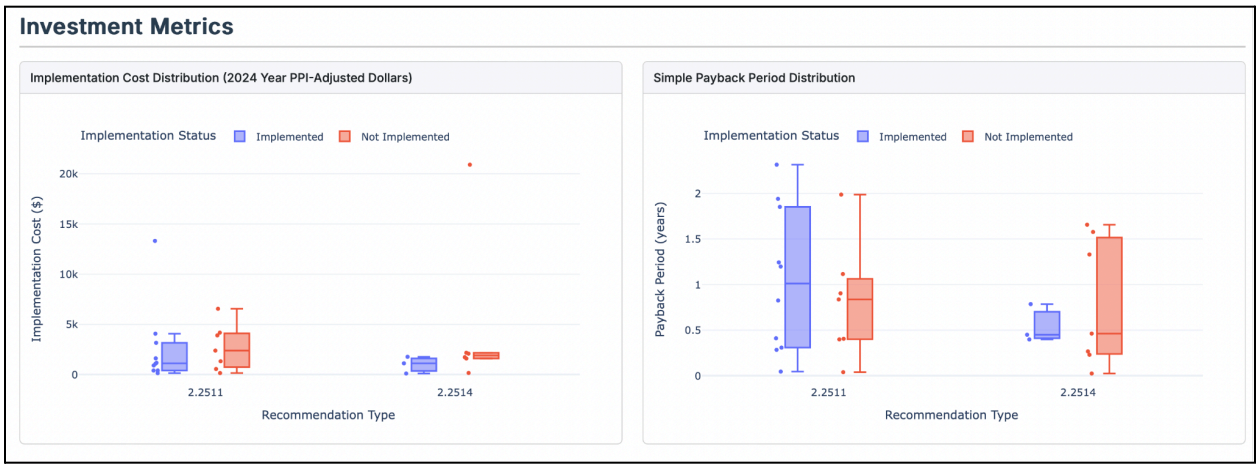
The following below is a list of boxplot titles, brief descriptions, and units currently available on the dashboard:

### Energy Metrics

- "Annual Electricity Savings Distribution" - Electricity saved per recommendation type in kilowatt-hours (kWh) per year.
- "Annual Natural Gas Savings Distribution" - Natural gas saved per recommendation type in million British thermal units (MMBtu) per year.
- "Annual Other Fuels Savings Distribution" - Fuels other than natural gas saved per recommendation type in million British thermal units (MMBtu) per year.

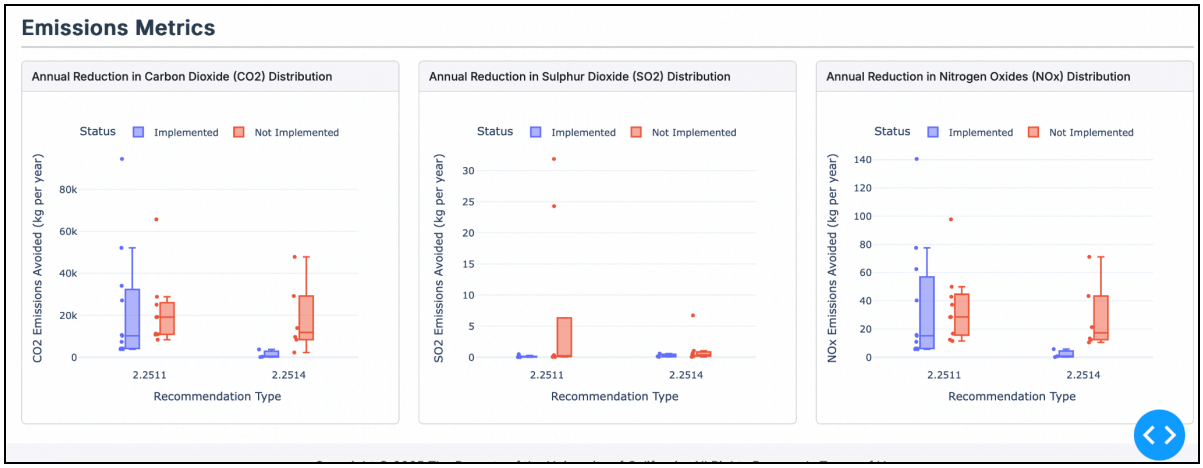

## Investment Metrics

- “Implementation Cost Distribution (2024 Year PPI- Adjusted Dollars)” - Implementation cost per recommendation type in United States Dollars (USD 2024$).
- “Simple Payback Period Distribution” - Payback period per recommendation type in years. Simple payback (years) is calculated as the implementation cost divided by the annual energy savings times energy cost.




## Emissions Metrics

- “Annual Reduction in Carbon Dioxide (CO2) Distribution” - CO2 emissions avoided per recommendation type in kilograms (kg) per year.
- “Annual Reduction in Sulphur Dioxide (SO2) Distribution” - SO2 emissions avoided per recommendation type in kilograms (kg) per year.
- “Annual Reduction in Nitrogen Oxides (NOx) Distribution” - NOx emissions avoided per recommendation type in kilograms (kg) per year.

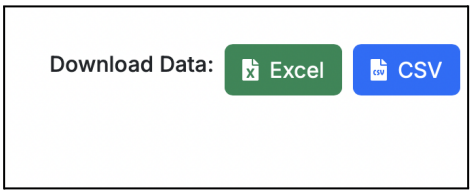



## Data Page

The Data page hosts the integrated dataset that the dashboard is utilizing at the time of deployment.

### Download Data

Users can download data by selecting the “Data” tab. The downloaded files are provided in either CSV or Excel format.




### Latest Data Update, Time of the Most Recent Update to the Dashboard

The dashboard is updated periodically. The latest update date is displayed at the top or bottom of the homepage or data tab (e.g., Last updated: May 2025). Please check this date to ensure you are working with the most current data available. As multiple datasets across several sources are used to generate the integrated dataset and the information on the dashboard, the dataset version information will be available under the “about” tab on the dashboard.

## Documentation Page

The Documentation page contains this User Guide, including a Frequently Asked Questions section listed below.

## FAQ (Frequently Asked Questions)

### Can I use this data for academic research or publication?

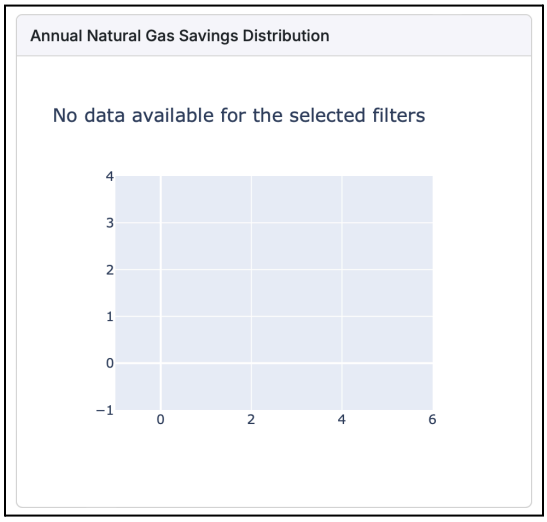
Yes, provided proper citation. Please refer to the dataset's license.

**Are cost savings in present-day dollars?**

Yes, all cost savings are represented in present-day USD and are adjusted for inflation and fluctuating costs of materials. This was done using the Producer Price Index (PPI), which measures the average change over time in the cost of materials. PPI data is published by the U.S. Bureau of Labor Statistics (BLS).

**Why are some data points missing or incomplete?**

Not all facilities report full implementation status, and some audits may lack complete metadata. It may also be expected that certain recommendations do not provide data on certain metrics. In such cases, visualizations will return the following message [“No data available for the selected filters”] and an empty chart.




**I think I found an error on the dashboard, who do I contact?**

For maintenance and upkeep of the dashboard, please contact Dr. Eric Masanet at emasanet@ucsb.edu. Please include screenshots of the error and a description.

# Data Update and Deployment Guide

## Dashboard Installation on a Local Machine

### Development Environment Setup

1. Install VS Code if you don't have it yet. https://code.visualstudio.com/
2. Install Anaconda if you don't have it yet.
3. Clone the repo from Git. In the terminal, run:

   ```
   git clone git@github.com:IndustrialEnergy/dashboard.git
   ```

4. Create a new conda environment. In the terminal, run:
   `conda env create -f environment.yml`
5. Activate the conda environment. In the terminal, run:
   `conda activate industrialenergy`
6. Update the file structure on your local machine by creating missing folders (e.g. some folders are in .gitignore and are not in the remote repo): refer to the section Repository Organization in this README.md.
7. Copy the latest available integrated dataset file from the Bren server directly apps.bren.ucsb.edu/capstone/industrialenergy/dashboard/data/final folder using SSH or from https://apps.bren.ucsb.edu/IE-Data/ to the data/final
   a. For SSH connection instructions, consult the guide Using VS Code to SSH into Bren Servers
   b. Alternatively, generate an updated dataset by running a data pipeline. Refer to section Data Update for step-by-step instructions.

### Running the Dashboard Locally

#### Locally without docker

1. Ensure that you are in the root folder.
2. In the terminal run:
   `python dashboard_app/server.py`
3. Run the dashboard in a browser on your local machine:
   http://localhost:3009/dashboard

## Locally in docker

1. Install docker desktop
   https://docs.docker.com/desktop/

2. Launch the docker desktop

3. Build the docker image

i. Navigate to the dashboard application root folder:

```
cd ../dashboard/
```

ii. Remove existing containers and volumes:

```
docker-compose down --volumes
```

iii. Rebuild images with --no-cache:

```
docker-compose build --no-cache
```

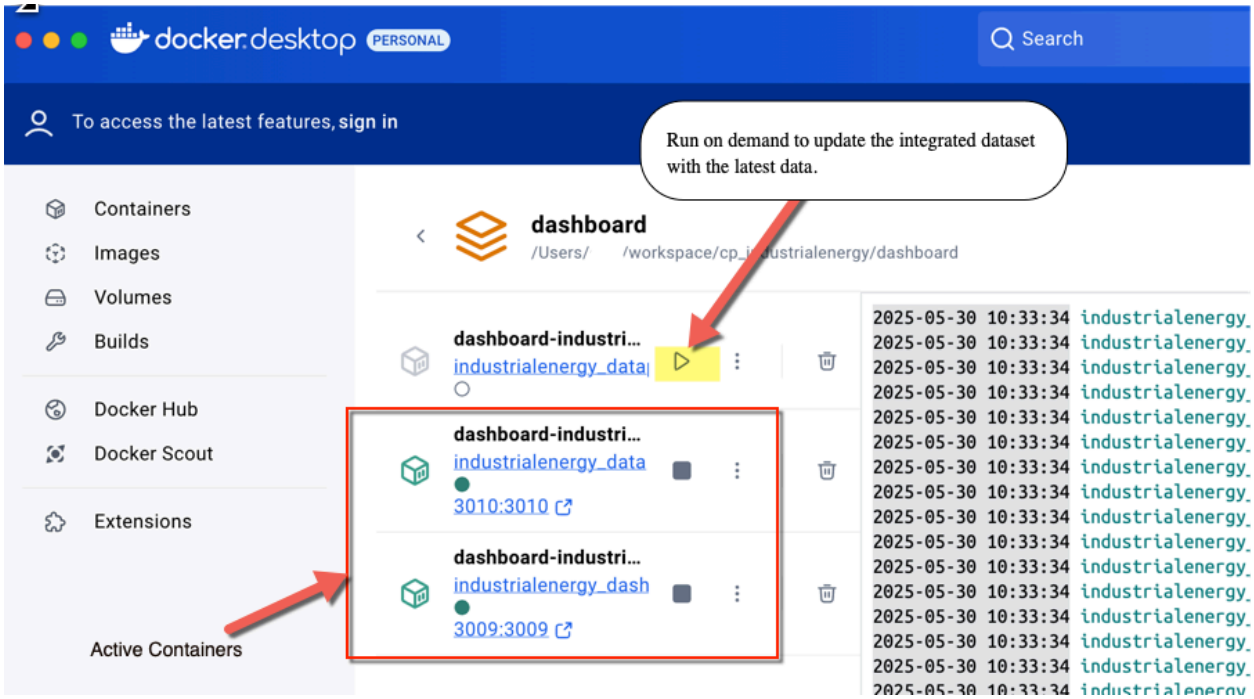
iv. Start docker containers with data and dashboard

```
docker-compose up -d industrialenergy_data industrialenergy_dashboard
```

4. Open the dashboard in a browser on your local machine:
   http://localhost:3009/dashboard

   Important: You can start the dashboard either using Docker or by running python dashboard_app/server.py directly—but not both at the same time. Doing so will result in a “port already in use” error.

5. You can view all docker containers and their status in the Docker desktop:




## Data Update

Before running the data pipeline, ensure that all required files are present, correctly named, and organized according to the folder structure described below.

- **raw/ folder**
  Include only the updated files in this folder. Files that haven't changed since the last update are already stored in the processed/ folder and do not need to be reuploaded.
  **Note:** When the data pipeline runs, files in raw/ will be moved to the archive/ folder, and the processed versions will be saved to the processed/ folder.

- **processed/ folder**
  This folder contains two types of files:

  1. Files automatically generated by the data pipeline
  2. Static files that do not change between updates (e.g., arc_descriptions.csv)
     **Important:** If any static file needs to be updated, manually edit it directly in the processed/ folder.

- **final/ folder**
  The fully integrated dataset is automatically saved here after the data pipeline is executed.

---

**Example of the correct folder structure:**

```
├── data
│   ├── Dockerfile
│   ├── archive
│   ├── final
│   │   └── iac_integrated.csv
│   ├── processed
│   │   ├── NAICS_SIC_Xwalk.csv
│   │   ├── arc_descriptions.csv
│   │   ├── ec_emission_factors.csv
│   │   ├── fuel_emission_factors.csv
│   │   ├── iac.csv
│   │   ├── naics_hierarchy.csv
│   │   ├── ppi.csv
│   │   └── sic_to_naics.csv
│   └── raw
│   │   ├── ARC_PPI_YYYY.xlsx   (e.g. ARC_PPI_2024.xlsx)
│   │   ├── IAC_Database_YYYYMMDD.xls  (e.g. IAC_Database_20250518.xlsx)
│   │   ├── annual_generation_state_YYYY.xls  (e.g.  annual_generation_state_2024.xls)
│   │   └── emission_annual_YYYY.xlsx  (emission_annual_2024.xlsx)
```

---

1. Ensure you have dashboard installed:
    a. If working on a local machine, complete steps 1-7 following instructions in the Configure the environment section to install the dashboard locally.
    b. If working on the Bren server, no additional installation is required. The dashboard is available in the following location on the Bren server: apps.bren.ucsb.edu/capstone/industrialenergy/dashboard/
2. Upload updated files to the **raw/ folder**
    a. Only upload the files that changed since the last data update.
    b. Ensure that the files have correct titles:

| Expected File Name | Example | Download URL |
|---|---|---|
| ARC_PPI_YYYY.xlsx | ARC_PPI_2024.xlsx | Google drive |
| IAC_Database_YYYYMMDD.xls | IAC_Database_20250518.xlsx | IAC website |
| annual_generation_state_YYYY.xls | annual_generation_state_2024.xls | EIA State Generation Excel |
| emission_annual_YYYY.xlsx | emission_annual_2024.xlsx | EIA Annual Emission Excel |

3. Navigate to the root folder
4. The data update script can be executed in Docker (**Recommended**) or locally
    a. To run in Docker (**Recommended**) - In the terminal run the following command:

```
docker-compose --profile pipeline run --rm industrialenergy_datapipeline
```

    b. To directly trigger a script - in the terminal run the following command:

```
Python tools/data_pipeline/run_pipeline.sh
```

5. The update process might take a few minutes to run. When completed, the updated file “iac_integrated.csv” will be generated and saved into the `/data/final` folder

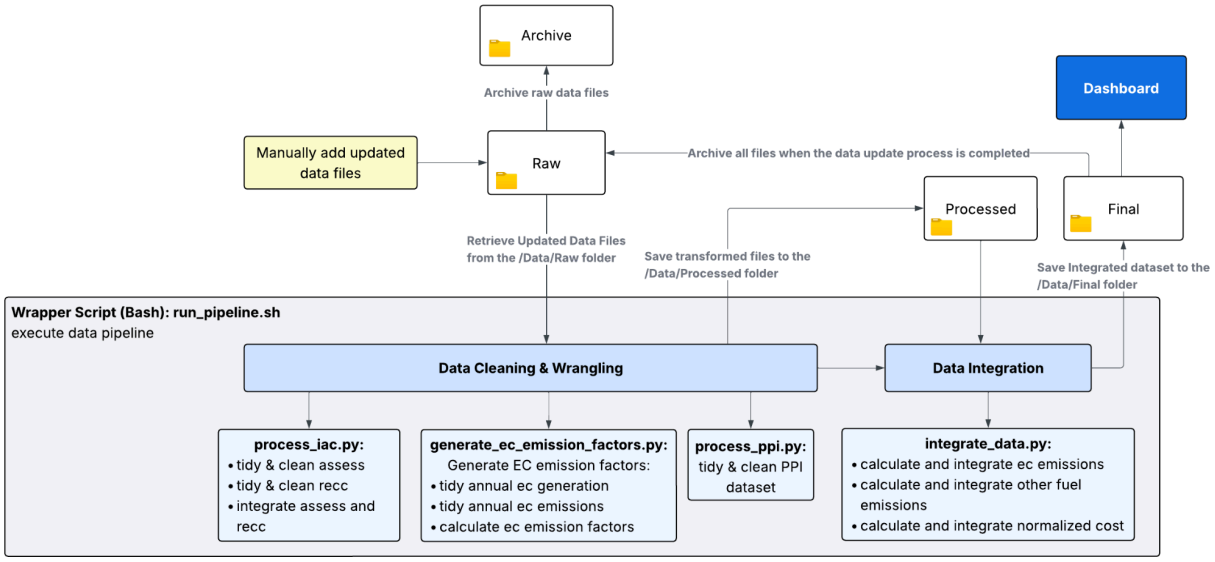



Figure 6. Data processing pipeline

**Deploying the Dashboard to Production**

**Note**: Deployment to the production environment is done using Docker containers.

1. Connect via SSH to the server apps.bren.ucsb.edu
2. Navigate to the dashboard root folder

apps.bren.ucsb.edu/capstone/industrialenergy/dashboard/

3. When in the server
    a. Open a terminal
    b. In the terminal run the following commands
        i. Navigate to the IndustrialEnergy application root folder:

        cd capstone/industrialenergy/dashboard/

        ii. Remove existing containers and volumes:

        docker-compose down --volumes

        iii. Rebuild images with --no-cache:

        docker-compose build --no-cache

        iv. Start docker containers with data and dashboard

docker-compose up -d industrialenergy_data industrialenergy_dashboard