1. Introduction

This project focuses on the acquisition, integration, and augmentation of electricity sector emissions and generation data from multiple sources, specifically the Australian National Greenhouse and Energy Reporting (NGER), Clean Energy Regulator (CER), and the Australian Bureau of Statistics (ABS).

2. Dataset Description

Data Sources:

- **NGER (National Greenhouse and Energy Reporting)**: This dataset provides emissions and generation data for the Australian electricity sector, covering the period from 2014 to 2024. This data was retrieved from the National Greenhouse and Energy Reporting website.
- **CER (Clean Energy Regulator)**: Data on proposed and planned large-scale renewable power stations across Australia. This dataset was sourced from the CER's market reports on large-scale renewable energy data.
- **ABS (Australian Bureau of Statistics)**: The ABS data focuses on population and economic statistics by regions. For this assignment, we selected the "population and people" data for analysis and integration into our dataset.

Data Preprocessing:

- **Cleaning**: Missing values (the data contains multiple values to define missing values such as empty string or dash "-") were handled by inputting the data with null values as this will make it easier to identify missing value. Data inconsistencies, such as varying date formats or categorical mismatches, were corrected. Irrelevant data such as empty facility names were also removed.
- **Data Type Conversion**: Ensured that numeric fields were properly formatted (e.g., converting strings to floats or integers).
- In the case of the ABS data, we have decided to retain the null values in the database for several important reasons:
    - **Data Integrity and Transparency**: By keeping the null values, we maintain full visibility into where data is missing. This helps in understanding the extent of the missing information and ensures that no data is inadvertently removed, which could lead to incomplete analyses. Having null values in the database provides a clear signal of where data is absent.
    - **Data Imputation and Handling Flexibility**: Removing the null values would require either imputing missing data or dropping rows entirely. Given the substantial number of null values across multiple columns, imputing this data could introduce biases or distortions, especially if the missing data is not missing at random. Keeping the nulls allows us to explore different strategies for handling the missing values without compromising the original dataset.

3. Data Exploration

Exploration Methods:

- Initial exploration involved assessing the completeness of the datasets and identifying any immediate issues like missing data, outliers, or format inconsistencies.

- Visual exploratory analysis was conducted on the data to observe trends and the distribution of the values.

Sample Visualization:

Figure 1 is a bar plot visualizing the distribution of power stations across different states. The plot shows the number of power stations in each state and highlights some key trends:

- **States with the Most Power Stations**: Western Australia (WA) has the highest number of power stations, followed by Queensland (QLD) and New South Wales (NSW).
- **States with Fewer Power Stations**: The Australian Capital Territory (ACT), Tasmania (TAS), and Northern Territory (NT) have significantly fewer power stations compared to other states.
- **Missing Data**: The "N/A", "-", and empty string category represents data that doesn't have a state classification or may have missing values in the dataset.

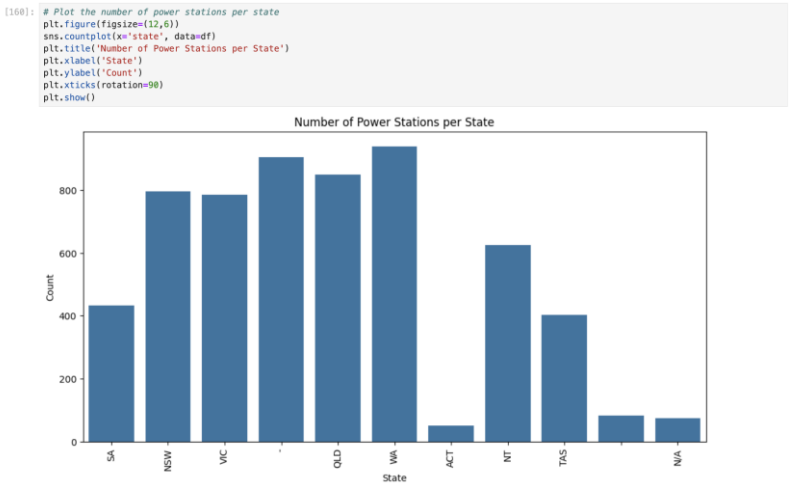

Figure 1. Number of Power Stations per State.

4. Data Augmentation

Geolocation Data:

To augment the dataset with geolocation information, we queried the geographic coordinates of all energy facilities (power stations) using the Nominatim geocoding API from OpenStreetMap. Each facility's location was retrieved based on its name/postcode, and state, and the latitude and longitude were stored for spatial analysis. If geocoding failed, a fallback mechanism used pre-fetched state-level coordinates, ensuring that every facility had some geographic data.

API Usage:

- We used the OpenStreetMap Nominatim API to retrieve the geographic coordinates.
- API calls were made for each power station to fetch the corresponding latitude and longitude values.
- A fallback mechanism was implemented for records that didn't return coordinates, we used the state coordinates, also obtained via the OpenStreetMap API.

Challenges:

The geocoding process initially took over 8 hours to run due to redundant searches using station names, and if that failed, the combination of station name and state, followed by state-level geocoding. This approach was inefficient, causing unnecessary API calls for each facility, especially when querying both station names and states multiple times, resulting in slow execution.

Solutions:

To improve efficiency, the process was split into two functions. The *get_state_coordinates* function checks the dataframe for unique states and queries the Nominatim API for state coordinates, returning a dictionary with state codes, coordinates, and osm_ids. This dictionary is then used in the main geocoding function *geocode_osm* to look up coordinates by station name/postcode and state. If the station's coordinates are not found, the function defaults to using the pre-fetched state coordinates, reducing redundant API calls and cutting the processing time from 8 hours to just 2 hours.

Recommendations:

- Enrich the dataset with the address of the stations to improve the accuracy of the geocoding.
- Implement batch geocoding to reduce the number of individual API calls and optimize processing

5. Database Design

Schema Design:

The database schema is designed with a star schema approach, which includes dimension tables and fact tables. This design provides an optimized structure for analytical queries and allows for easy aggregations. The primary dimension tables store descriptive data related to the energy facilities, fuel types, locations, and time, while the fact tables store quantitative metrics such as energy production, CO2 emissions, and population data. The diagram of our schema is provided in Figure 2 of the appendix.

Table Design:

- Fact tables:
  - *fact_nger_metrics*: Key metrics such as energy production and emissions data for each facility, tracked over time periods.
  - *fact_population*: Population-related metrics like estimated resident population, population density, and age demographics by region and year.
- Dimension tables:
  - *dim_nger_facility*: Information about the energy facilities, including type, fuel, location, and grid details.
  - *dim_cer_station*: Information about the clean energy power stations, including fuel, location, production capacity and status details.
  - *dim_fuel*: Fuel types used by the energy facilities.
  - *dim_location*: Geolocation data for each facility, including latitude, longitude, and geometrical information (spatial data).
  - *dim_state*: State-level data, including state codes and names
  - *dim_time*: time-related attributes such as year, month, week, for effective time-based analysis.

Database Design Justification:

- A normalised design was chosen, specifically using a star schema, to optimize for query performance in analytical environments and remove redundancy in storing the data. This approach reduces the need for complex joins and speeds up aggregation queries, such as calculating total energy production or emissions over time, fuel types, or states.
- Spatial data was stored using PostGIS (PostgreSQL spatial extension), enabling spatial queries on the geographical distribution of power stations and facilitating location-based analysis of energy facilities.
- The time-related data was stored in a *dim_time* table to ensure time-based aggregation and analysis. The table provides a separate dimension for time-related data, thus minimizing duplication and ensuring that time-related queries can be executed efficiently.
- The design follows the relational model with foreign keys linking the fact and dimension tables. This maintains data integrity and enables easy relationships between different entities, such as connecting power stations to specific locations or fuel types.

Data Transformation:

In our data transformation process, we focused on cleaning and structuring various datasets into the format we designed for our schema, ensuring the data could be efficiently loaded into a PostgreSQL database. This involved standardizing columns, consolidating data from multiple sources, and ensuring that each table was in a format suitable for analysis and reporting. We also ensured that the data was properly prepared for spatial queries by converting geolocation data into PostGIS-compatible formats.

One of the main challenges we encountered was merging datasets from different sources, which required maintaining consistent keys and proper indexing across the data. To overcome this, we assigned unique IDs to each entity, ensuring that each row had a distinct identifier. While some tables could be managed using a single field as a unique identifier, others required more complex composite keys. These composite keys were created by combining multiple fields to form a meaningful, unique identifier for each row, which was essential for maintaining the integrity of the relationships between different tables.

Additionally, we made sure to account for any potential duplicates or inconsistencies in the data by carefully grouping and aggregating rows. This approach helped eliminate redundancy and ensured that the data was both accurate and well-organized, further enhancing the overall quality and integrity of the dataset.

6. Team Contributions

- Muhamad Naufal Azwar Iftikar (SID 540908237): Led data acquisition, data integration, and data cleaning. Led code refactoring.
- Naufal Rauf (SID 540844821): Designed the database schema, performed data augmentation, and created and inserted the data into the schema.

7. Appendix

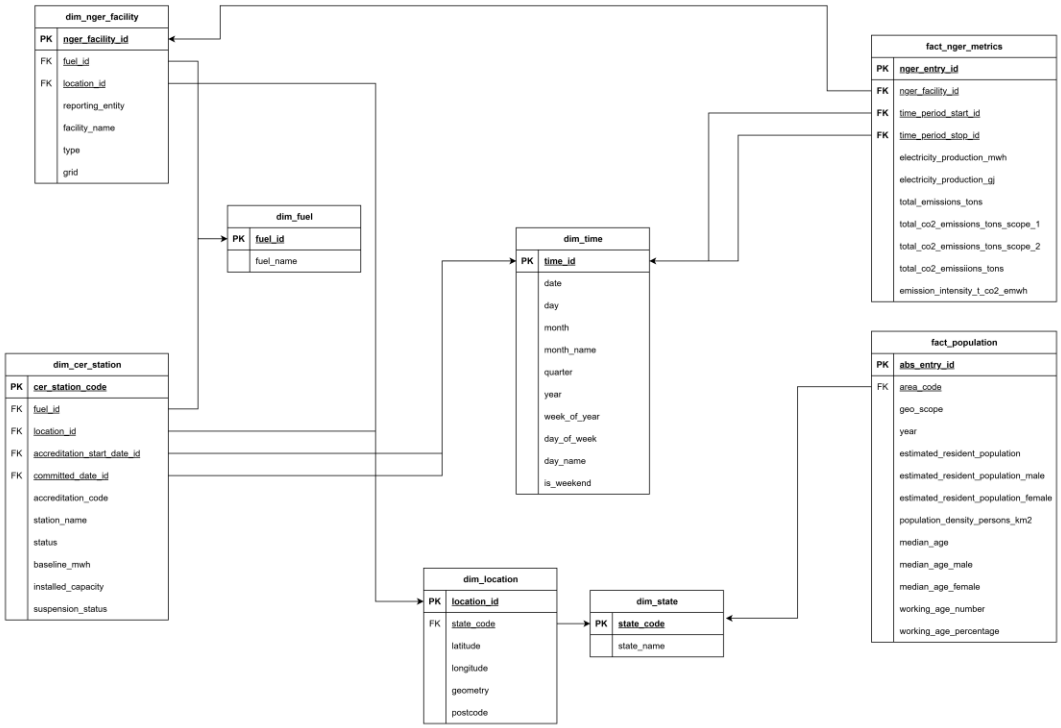

Figure 2. Diagram of the schema of our energy database