Stephen Xu

August 7, 2023

GIS Comparison and Aggregation Report

**Task Description**

This Python project aims to compare Generator Projects as mentioned in ERCOT’s GIS Reports for the last two months. In particular, the project finds the difference between the projected COD (Commercial Operations Date) between corresponding projects and outputs that data to a CSV.

Additionally, the project has the functionality to aggregate the previous four GIS reports into one large CSV.

**Solution Design**

1. Script Location: *\\pzpwcmfs01\CA\11\_Transmission Analysis\ERCOT\101 - Misc\CRR Limit Aggregates\Python Scripts\Wind and Solar Aggregation*
2. Output Location: *\\pzpwcmfs01\CA\11\_Transmission Analysis\ERCOT\101 - Misc\CRR Limit Aggregates\Data*

This can be adjusted manually by editing the *output\_path* variables in each of the Python scripts.

1. Running *GIS\_Comparator.py* compares the latest two GIS reports and determines the change in projected COD between the two files.
2. Running *GIS\_Aggregator.py* aggregates either the latest four files or all historical data going back to 2019.
3. **Important**: The *GIS\_Aggregator* script contains a version variable on line 13 that changes the overall behavior of the program. If the version is set to 1, all historical data will be aggregated, but if the version is set to 2, only the previous four will be.

**Important Prerequisite Information**

The Python scripts for this project read the Excel sheets located at [*\\pzpwcmfs01\ca\11\_Transmission Analysis\ERCOT\02*](file:///\\pzpwcmfs01\ca\11_Transmission%20Analysis\ERCOT\02) *- Input\Generation\New Entrants\ERCOT Queue*.

The Excel sheets in this folder all contain images, which the Python pandas library can have difficulties reading. Be sure that the **openpyxl** package is at least version **3.1.2.** You can verify if the version on your computer satisfies this requirement by creating a new Python script, importing openpyxl and printing “*openpyxl.\_\_version\_\_*”*.* If you do not meet this requirement, follow the instructions below:

1. Open the Anaconda Prompt from the start menu.
2. Type in the following command: *pip install openpyxl==3.1.2*
3. Press Enter and see if the installation was successful. If you get denied by a permission error, restart at step 1 and run the Anaconda Prompt as an administrator. Use an SA account (or ask IT) if prompted.
4. If you cannot get steps 1-3 to work, you can create a new conda environment in a folder that you have write access to. Type the following command:  
    *conda create --name {env\_name}*Be sure to replace *env\_name* with your desired environment name.
5. From here, install all necessary packages to that environment via *conda install* *{pkg\_name}.* In particular, you will need the command

*conda install openpyxl=3.1.2*

You will also need the Pandas library. I don’t believe the Pandas version is too important here – the default one should suffice.

1. Be sure to redirect the IDE you use to develop Python to run these scripts using the new conda environment.

Sample Output

