## Aurora Load Script Documentation (updated October 16, 2020)

Author: Sumin Wang ([sumin.wang@ethree.com](mailto:sumin.wang@ethree.com))

# 

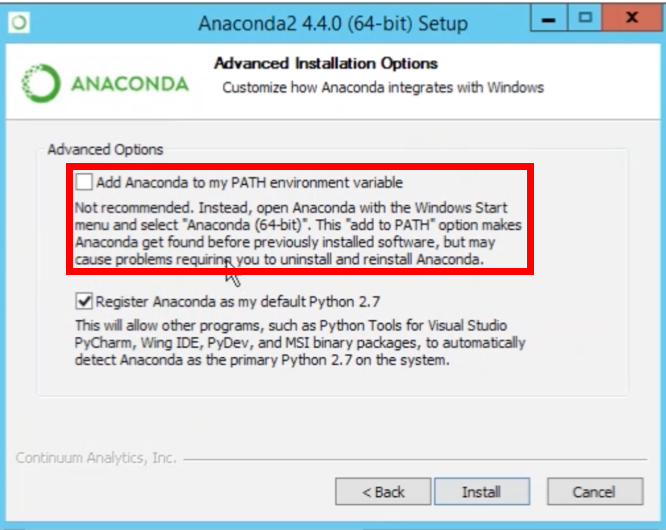
# Setting up the python environment

Note that the Aurora Load Script can also be run with python environment. If you have this already you can skip this step

To run the model a specific version of python and certain python libraries must be available. To set this up we use an environment which is a set of libraries and conditions required to run a program, if you’re interested in learning more see [this excellent guide](https://www.freecodecamp.org/news/why-you-need-python-environments-and-how-to-manage-them-with-conda-85f155f4353c/).

To set up the environment's anaconda must be installed on your PC first. You can find this here: <https://www.anaconda.com/distribution/>

We recommend installing the python 3.7 version and following the default installation instructions. Please make sure to select “add Anaconda to my PATH environment variable” during the installation. If you have already installed Anaconda but forget to select this advanced option, you could check section 2.5 and 2.6 of this [E3 Python Installation Guide](https://ethreesf.sharepoint.com/:w:/s/Models/EaKH2ZWEcaxCmQltzaAXfPQBplGUwmFULDjGiOSb9hp87g?e=JOcLGc) and learn how to add Anaconda to PATH manually.



# Load Inputs

The current structure of the script is split into **3 folders and 1 input Dashboard**:

* **Input Dashboard**: Load Input.xlsx. Used mainly to specify load allocation and key assumptions
* **Raw\_data** stores raw inputs into the script
  + Four Scenarios\_Original.csv: the raw load data coming from PATHWAY team
  + SH\_Shape\_PATHWAYS\_12112019.csv: latest version of shapes for electrification load
  + Aurora\_raw.csv: default aurora load
  + Resources disaggregated column names.csv: column names to generate Resources Aggregated table that goes into Aurora
    - Other files are legacy files that might be deleted in the future. Could ignore now
* **Calc** stores intermediate calculation coming out of the script. It serves for QAQC purposes
  + Aurora\_inter\_area\_relationship.csv: the ratio used to split up load in “Census Division” level to “Demand Area” level. Depending on the allocation is defined in Load Input.xlsx, the sum of the ratio for one larger region should be equal to 1.
  + Aurora\_load\_peak\_relationship.csv: the ratio used to calculate peak load using annual load. The ratio should be consistent with the ratio in Aurora\_raw.csv
  + Four\_scenarios.csv: Four Scenarios\_Original.csv benchmarked with EIA actual load values in 2019
  + [Electrification load type]\_load\_by\_area\_[SCENARIO]\_[DATE]\_[TIME].csv: electrification load split into areas by scenarios.
* **To\_Aurora** stores all the inputs that need to be copied into Aurora. It’s defined by corresponding table name in Aurora and scenario.

# Scripts

Only two scripts are important to use to generate baseload and electrification load:

* **load\_temp\_base.py** 
  + this is to get baseload
* **load\_temp\_elec.py**
  + this is to get electrification load

After running both scripts, the user should be able to get the results under “to\_aurora” folder.

Eia\_benchmark.py is used to benchmark PATHWAY forecast with the most updated EIA historical load output. We already used this script to benchmark PATHWAY load with the 2018 EIA forecast if you’re using the default inputs. But this is going to be irrelevant if you use your own load forecast.

# Methodology

## Base Load

Base load (in TWh) is the sum of “Building Other” and “Industry” load coming out of the PATHWAYS inputs. They come in at the “Census Division” level. The script allocates load in “Census Division” level to smaller “Demand Area” level that Aurora uses in MWh.

The allocation is done by assuming:

* Base load input is on retail level that needs to be converted to wholesale level by accounting for transmission loss
* The annual load relationship across “Demand Area” are consistent with the **2018 default load relationship in Aurora**
* The peak to annual load relationship across “Demand Area” are consistent with the **2018 default load relationship in Aurora**

## Electrification Load

Electrification load is comprised of five types of load: space heating, water heating, heavy duty vehicle (HDV), light duty vehicle (LDV) unmanaged, and LDV managed.

In Aurora, electrification load is modeled as a capacity resource. Its dispatch is determined by the capacity of the resource and 8760 stream of capacity factors.

* **Capacity**:
  + Using the same methodology as Base Load, electrification load in TWh at “Census Division” level is split into more granular “Demand Area” load in MWh
  + For each demand area,
    - Capacity = - annual load \* max of the normalized shape
    - Capacity is negative because they’re modeled as a generation resource but they actually consume energy rather than supply energy
  + PATHWAY team provides the normalized shape that sums up to 1. Hourly load is normalized by total annual load.
* **8760 stream of capacity factors**:
  + The 8760 capacity factor is used in Aurora to determine the electrification load will be dispatched.
  + Capacity factor at t = annual load at t/maximum annual load

LDV load is further split into unmanaged (LDVU) and managed load (LDVM).

LDVU load = LDV load \* (1- % of managed charging)

LDVM load = LDV load \* % of managed charging

% of managed charging is specified under Load\_Input.xlsx

Currently, unmanaged and managed charging load are both treated as a generation resource with negative capacity like space heating load. EV charging load shapes are from EVGrid team.

The following information is not relevant for the most recent version of the script. This is how we implement managed charging in the past

Unmanaged charging load is treated as a generation resource with negative capacity and fixed shape like space heating load because they’re not dispatchable. Managed charging load is modeled as a battery and could be dispatched. Therefore, it’s modeled with a storage unit. A storage unit will come with both power capacity (MW) and energy capacity (MWh) attributes.

Energy capacity of LDVM load = power capacity of LDVM load \* duration

LDVM load

* Discharging:
  + Normalized shape
  + Capacity shape
    - Negative because we model the resource as a storage unit. We need to set it as a negative number to let the energy flow out. Discharging shape should look the opposite of the charging shape.
* Charging
  + The shape shows charging availability shape, which is different than the other shapes. It tells you whether a storage unit is ready to charge or not. Charging availability should be complementary to discharging shapes. Charging availability to adjust the storage capacity to represent when you’re driving, you can’t charge.