Currently the application is able to extract data from the ASR and merge the GPON csv data into a combined CSV file that allows you to search through each customer and see their data usage. This needs to have a mechanism of dealing with historical data as well, not just the immediately extracted information. This application currently keeps track of historical records by exporting an entire list of customers (currently only GPON 7/6/18), every 6 hours (Or specified time).

The application WILL not deal with front end user issues, it is simply designed for extraction of customer data. This means that anything like multiple searches, global average data usage and other features may not be implemented with this script, as it is not in the scope of it. A frontend may however use this script to extract the appropriate data to build generalized usage statistics from all combined customers or use it in such a way so as to search for more than one customer at a time.

The application will allow you to search for more than one argument at a time, this means that for example, you could search for a customer’s name as well as their ONT or likewise. This allows us to more easily deal with customer’s that have overlapping names or OTHER overlapping information.

# Specifications

* Use camel case for methods and variables: thisIsAnExample
* Class names should also use camel case, BEGINING with uppercase: ClassNameExample
* Globals should always use all caps and no spaces: ALLCAPSNOSPACES
* Do NOT use ALL lowercase names: thisisalist
* Attempt to divide up methods into smaller chunks that can easily be used elsewhere in the application.
* Comment as much as possible, if it seems like it’s confusing, (ATTEMPT) to put a semi meaningful comment explaining what the function or process is doing.
* DON’T make thing’s unnecessarily long, this means if for example, you’re iterating through a list to extract variables, use the bracket feature [], DIRECTLY in the process you’re going to use the data in, DO NOT put them in a separate variable UNLESS it’s necessary in more than one place.

Script that can sort through GPON and Active E CSV files that can search for; customers, VLANs, or other data. Results should yield the data usage history of a specified customer.

Active E still needs to be parsed through, should be a separate module from GPON. This WILL NEED a standardized way of extracting the GPON and Active E data, so the application can parse the data appropriately.

Active E will have to have a more robust approach to extract the data from it. The data needs to be translated from which port number, shelf, location, etc, into the appropriate VLAN tag, before we can merge it with the appropriate ASR data.

This application is a backend, a GUI will NOT! be built for this application. This script is ONLY for extracting the GPON, ACTIVE E data into an appropriate format that then can also be searched through with another script.

Application does NOT need to search for multiple customers, ONTS, IDS, indexes, VLANS, etc. at a single time!!! This can be handled later with a front end that simply calls the script to search for each individual customer!

When searching through historical CSV’s (an individual walk through of the ASR) we will have a function that combines the users octet (up and down data usage) into a single list so that we have a nice curated list of historical data usage on an individual customer.

When dealing with multiple customers with the same name, we will need to either figure a way to deal with this issue or prompt the user for more input to select the correct customer. I may be able to mitigate this issue if I can figure out some way of going through the CSV’s ahead of time and figuring out what customers have the same name and going from there to curate them further.

# # OLD COMMENTS

### Can also sort through all customers and give an average of total usage? (Unsure if this would be useful at this time. 7/6/18) ### THIS WOULD NOT BE DONE IN THIS SCRIPT, A FRONT END WEB APPLICATION CAN USE THIS SCRIPT TO EXTRACT THE DATA REQUIRED FOR THAT AND GO FROM THERE. (7/13/18)!!!