STANDARD OPERATING PROCEDURES
FOR
CALIFORNIA AIR RESOURCES BOARD LOGGER
(CARBLogger)

AQSB SOP 605

Second Edition

MONITORING AND LABORATORY DIVISION

April 2020

Disclaimer: Mention of any trade name or commercial product in this standard operating procedure does not constitute endorsement or recommendation of this product by the California Air Resources Board. Specific brand names and instrument descriptions listed in the standard operating procedure are for equipment used by the California Air Resources Board's laboratory. Any functionally equivalent instrumentation is acceptable

CALIFORNIA
AIR RESOURCES BOARD

Approval of Standard Operating Procedures (SOP)

Title:	CALIFORNIA AIR RESOURCES BOARD LOGGER (CARBLogger)
SOP:	AQSB SOP 605, Second Edition
Section:	Operations and Data Support Section
Branch:	Air Quality Surveillance Branch (AQSB)
Division:	Monitoring and Laboratory Division (MLD)
Prepared by:	Operations and Data Support Section

Reviewed by:

Manisha Singh, Ph.D., Chief
Quality Management Branch

5/4/2020
Date:

Approval by:

Reggie Smith, Manager
Operations and Data Support Section
Air Quality Surveillance Branch

23 April 2020
Date:

Kathleen Gill
Kathleen Gill, Chief
Air Quality Surveillance Branch

05/06/2020
Date:

TABLE OF CONTENTS

FIGURES

TABLES

APPENDICES

REVISION HISTORY

Edition	Release Data	Changes
First	October 2017	New Document
Second	April 2020	ADA Remediation

LIST OF ACRONYMS

AQSB	Air Quality Surveillance Branch
ASD	Administrative Services Division
CalEPA	California Environmental Protection Agency
CARB	California Air Resources Board
CentOS	Community Enterprise Operating System; a Linux / Red Hat variant.
CL	CARBLogger
DAS	Data Acquisition System
DMS	Data Management System
FTP	File Transfer Protocol; refer to the FTP Server in this document
GMT	Greenwich Mean Time
GUI	Graphical User Interface
IZS	Internal Zero-Span
LST	Local Standard Time
NTP	Network Time Protocol
ODSS	Operations and Data Support Section
OIS	Office of Information Services
OS	Operating System
PST	Pacific Standard Time
QA	Quality Assurance
QC	Quality Control
SFTP	Secure File Transfer Protocol; refer to the SFTP Server in this document
SOP	Standard Operating Procedure
UPS	Uninterruptible Power Supply
U.S. EPA	United States Environmental Protection Agency

1.0 GENERAL INFORMATION

1.1 Introduction:

The CALIFORNIA AIR RESOURCES BOARD LOGGER (CARBLogger) is an open-source digital data acquisition system developed by Operations and Data Support Section (ODSS) staff, for many uses in the continuous ambient air monitoring program hosted by the Air Quality Surveillance Branch (AQSB) of California Air Resources Board (CARB). Originally, the system was designed to provide minute-based diagnostic data for instrumentation used in the ambient air monitoring sites. As it proved to be consistent and reliable, the system was further developed to serve as a full data acquisition system, and allowed AQSB to fully utilize the Data Management System (DMS) for supporting the AQSB ambient air monitoring program.

CARBLogger (CL) utilizes a variety of open-source applications to collectively acquire digital data, allow for secure remote access, screen diagnostic data, provide e-mail notifications, and flag data for different operating states from a variety of air monitoring instruments used in the monitoring network.

The purpose of this standard operating procedure (SOP) is to document installation, operation, maintenance and troubleshooting procedures for the use of the CARBLogger system. Station operators who may need to modify or customize their data logger for specific sites should contact staff of the Operations and Data Support Section.

Note: This SOP references CARBLogger version 1.2.

1.2 Principle of Operation:

The following process diagram (see Figure 1.1) demonstrates the general relationship between air monitoring instruments and the various shell scripts and stored procedures run from the CARBLogger.

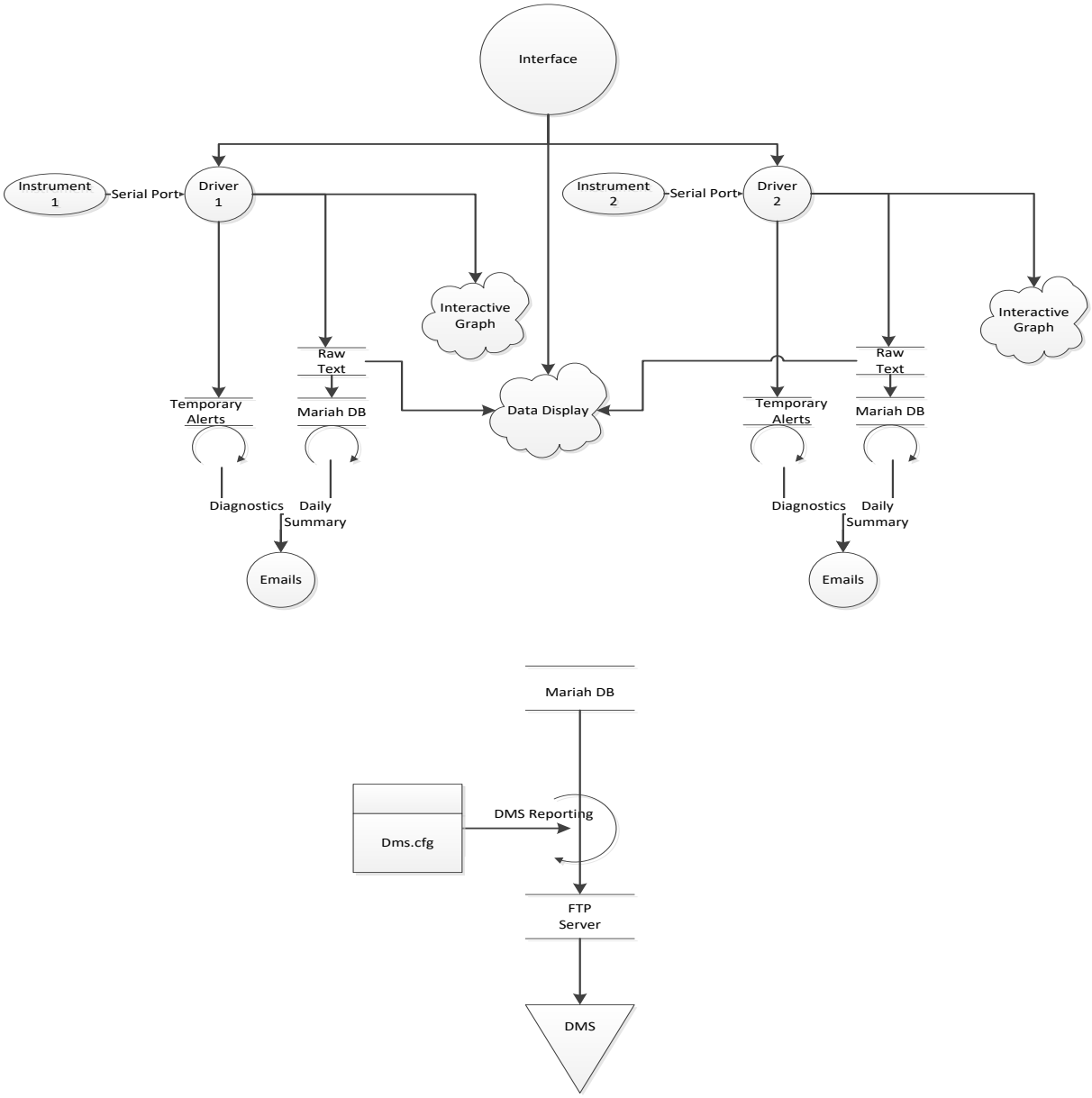


Figure 1.1: CARBLogger Data Processing Diagram.

1. CL acquires "raw data" from air monitoring instruments via RS232, RS485, or Ethernet connections, using serial, file sharing, JSON, http, or Modbus protocols.

2. The "raw data" stream is time-stamped using Network Time Protocol (NTP). This timestamp corresponds to the time at which the data point was acquired and ensures data collected from instruments are time synchronized with CL. In addition, this raw data is also modified for parsing by the CL user interface for direct display and for DMS reporting.

3. CARBLogger calculates the correct DMS time for data storage using the CL timestamp, minus the offset stored in the dms.cfg file. For example, if the CL timestamp is 8:50 and the offset is set to 1 minute, the DMS will receive 8:49 as the correct recording time.

4. As data are being collected by CL and written to disk, each instrument's diagnostic data stream is being screened in parallel using criteria established by AQSB, as indicated on each instrument's monthly Quality Control (QC) maintenance check sheet. CL drivers note any diagnostic data outside of acceptable QC criteria for each instrument's respective parameters and append any violations to the CL's errors file.

5. Twice a day, instrument diagnostic data and errors are summarized and emailed to the station operator, calibrator, and site supervisor. These emails are referred to as diagnostic email alerts.

6. CL also stores the collected data into its onboard database.

7. CL database has custom stored procedures which interact with the dms.cfg file, display files, and its stored data, to produce op-coded data files in a native DMS format. These data files are commonly referred to as ". MIN" (dot min) files. The timestamps used for the ".MIN" files are based on the NTP timestamps discussed above. In addition, ".MIN" files are transferred hourly to DMS through the CARB's SFTP server.

8. Each day CL additionally summarizes the previous days "raw data" file and daily one minute records, and emails these files to the station operator and site supervisor. These are referred to as raw data report and data extract report, respectively.

9. CL also allows remote access to connected instruments 24 hours a day when the network is correctly configured.

1.3 CARBLogger Limitations:

For clarity, we delineate what CARBLogger does and does not perform for DMS.

1. CL assigns Op codes to each instrument minute/hour value based on the state of the calibrator, the time of day, the status chosen by the operator, the time stamp applied to the data, and the manifold the instrument is connected to. QC codes are assigned by DMS using, a) Op to QC code mapping, b) AutoQC validation criteria, or c) manual QC assignments.

2. CL does not average minute values into hourly values. Hourly data displayed on DMS are aggregated by DMS from the ingested minute values for each instrument.

3. The current version of CL does not assign an Op code (flag) to minute data points based on diagnostic data results. The Op code assignment is solely based upon the system's calibration state, or the operator's choice of an enable/disable channel state selected from the CL user interface. The exception to this rule is the BAM-1022 driver, which flags data invalid if error codes are detected. The current BAM-1022 firmware uses a full scale data value to indicate an error code, but integrates the value into the hourly average. The artificially high values are invalidated by the driver.

4. For sites using the Environics 9100 gas calibration system, a custom SQL Server stored procedure changes the QC codes for the first number of minutes of each calibration step that CARBLogger reports as being in a calibration phase. This is not the case, however, for sites using the Sabio4010, APIT703, or APIT400 internal zero span gas calibration systems. This number of minutes is dependent on the instrument being calibrated.

5. DMS calculates calibration values using contiguous Op code assignments of zero, precision, or span made by CL. DMS will calculate incorrect calibration values if the flagging is incorrectly assigned by CL.

1.4 Installation Constraints:

1. CARBLogger must be installed in an environmentally controlled location with clean power suitable for the platform it is running on.

2. As with any electronic equipment, CARBLogger must be electrically grounded and connected to a suitable power supply unit. If not, instrument connections which are subject to noise and bad grounds (i.e., RS-232, Ethernet, or TCP/IP) may cause intermittent connection and data outages.

3. The Dell R620 (CARBLogger) produces a +/- 20% watt draw on its 490-watt power supply, giving an upper end wattage requirement of 588 watts, another 20 watts for the screen, or roughly 5 amps at 120 VAC.

4. Care should be taken when using small circuits or power cords, as a single moderate draw device such as a refrigerator, space heater, or window air conditioner can cause power problems.

5. CARBLogger version 1.2 requires an Ethernet connection with a static IP address and port forwarding to operate normally.

6. CARBLogger collects digital data exclusively, hence, there are no status contact or analog output data collection capabilities in the CL system.

7. CARBLogger can serve as a remote terminal server to allow remote access to control and modify station instruments; advanced network configuration may be required, which is beyond the scope of this documentation.

8. CARBLogger cannot report data directly to data clients nor is it designed to do that.

1.5 Safety Precautions:

CARBLogger hardware should be installed in a well-grounded, power conditioned environment and should be kept clear of any water and/or precipitation.

Operators should adhere to the hardware manufacturer's precautions for the specific CARBLogger platform being used.

Any precautions taken with a desktop computer or server should also be applied to the CARBLogger.

1.6 Personnel Qualifications:

Staff should be trained and familiar with basic air monitoring principles and procedures prior to operating any air monitoring equipment. Staff should complete any required safety training before operating any air monitoring equipment and working in the field. Staff should review this SOP, the instruments' operating manuals, and complete any instrument related and air monitoring operations training required by their section manager prior to operating the CARBLogger.

2.0 INSTALLATION PROCEDURE

2.1 General Information:

Installation of CARBLogger should be similar to installation of any ambient air monitoring instrument used by AQSB. CARB monitoring practices state that the instruments should be installed in a stable, temperature controlled environment, between 20° C to 30°C. Care should be taken to install the instrument in a standard 19" instrument rack such that it can be accessed for maintenance, repair work and troubleshooting. The standard 19" instrument rack should be bolted to the floor and properly grounded. CARBLogger should be installed such that operators can attach a monitor, keyboard, and mouse to access the graphical desktop.

2.2 Equipment & Materials:

The following list of equipment is based on the installation of a new site, where no previous CARBLogger has been installed. If one needs only to upgrade or replace an existing CARBLogger, transfer kits are available on request.

- One uninterruptible power supply (UPS), and if available, one power conditioner.
- Two power cables; CARBLogger has two power sources.
- One set of USB-based keyboard and mouse.
- One LCD monitor screen.

Other materials needed to install the CARBLogger chassis in the air monitoring station rack can include any of the following:

- Wooden planks with hardware brackets.

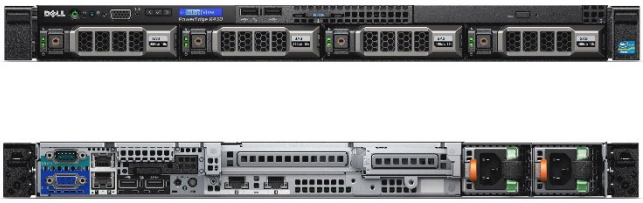
Figure 2.1: Wood Plank Mount

- Environmental equipment shelves.
- One (or two) PCIe serial cards and accompanying 8 port DB9M fan-out mini cable(s) (product code 04001620).
- At least one serial cable per instrument.
- A DSL modem, router, or other source of Ethernet connectivity.
- One small Philips screwdriver, one small flathead screwdriver, and a flashlight.

2.3 Available Form-Factors:

Inspect the CARBLogger chassis and confirm that it is free from any physical defects. At the time of this writing, there are four different hardware platforms that CARBLogger is currently being used on (as shown in the table below):

Table 1: Different CARBLogger Form Factors

Platform	Images
Dell R430/R440	
Dell R620	
Dell T610	 

Platform	Images
HP8000	

Note: Per CARB's IT equipment policy, each CARBLogger is issued an instrument barcode that is attached to the outer shell. Please record this barcode for inventory management.

Locate and record the CARBLogger serial number. In case there is a need for technical support from the manufacturer (Dell) in future, this serial number is required for support approval. For technical support on the HP8000 platform, please contact the Office of Information Services (OIS) directly.

2.4 Installation Inspection:

CARBLogger should be placed inside an environmentally-controlled air monitoring station and secured into the equipment rack. The CARBLogger should be well grounded, connected to a UPS and a conditioned power supply. Repetitive power outages and poor power conditioning can and will cause hard disk failures and file system corruption.

Installation should be straightforward. Any precautions for installation taken with other electronic equipment should also be applied to the CARBLogger installation.

CARBLogger is configured in the ODSS shop before being shipped to the field. The PCI serial card(s) are installed and tested by CARBDMS TEAM staff. If there is need to replace a faulty serial card in the field, CARBDMS TEAM will issue a separate set of instructions/procedures for field component replacement.

Make sure the 8-port cable and instrument serial cables are connected securely. Use a zip-tie/plastic brace to hold the weight of these cables to the chassis/rack.

Avoid free-hanging cables from the back of CARBLogger.

2.5 CARBLogger Connection:

Connect the monitor screen, keyboard, mouse, and the two power cables to the CARBLogger peripheral ports on the back of the system.

Note: Tie-downs and/or plastic braces should be used to secure all cabling to the CARBLogger chassis and relieve any strain placed on the cables.

Connect each instrument to one of the available serial ports from the 8-Port serial cable, starting with the cable labeled #1.

Once all the instruments are connected to the CARBLogger, operators may power on the CARBLogger.

Figure 2.2: Securing Cables to the CARBLogger Chassis.

Please make note of which instrument is connected to which serial cable number, as you will need this information later in the instrument configuration section.

If you require more than eight serial cables, it is possible to add additional serial connections by acquiring another serial card and making a slight change to the CARBLogger configuration file. Alternatively, you can use the remaining serial ports available on the chassis.

3.0 USER INTERFACE

3.1 Introduction:

The CARBLogger user interface was developed by ODSS to provide users a simple way of controlling the underlying data acquisition system. It provides users a means to add, remove, start, stop, and configure all instrument drivers and site settings. It also provides users a manner to flag data into a different operational state, i.e. offline, maintenance, or calibration, etc.

Each CARBLogger menu consists of a numbered list of usable options, each of which will be addressed here.

3.2 User Interface Home:

The user interface home refers to a scripted menu system which displays the files and folders located under the "*/home/aqdms/CARBLog*" (a.k.a., CARBLog) folder.

This interface allows users to control the various drivers and reporting processes available on the local machine without needing to use or understand the Linux command prompt and related files.

Interface navigation begins at the home screen/menu. Figure 3.1 shown on the following page is the home menu that starts when the CARBLogger is turned on.

Additional menus can be accessed by selecting the number preceding it on this screen, each of which will be addressed in the following subsections. The following topics will be discussed:

- Starting/stopping instrument(s) or the CARBLogger.
- Adding new instrument to data logger.
- Editing existing instrument settings.
- Delete existing instrument from data logger.
- Enabling/disabling monitor channels.
- Showing data to display monitor.
- Editing/resetting station details.
- Generating raw data extract for manual DMS backfill.

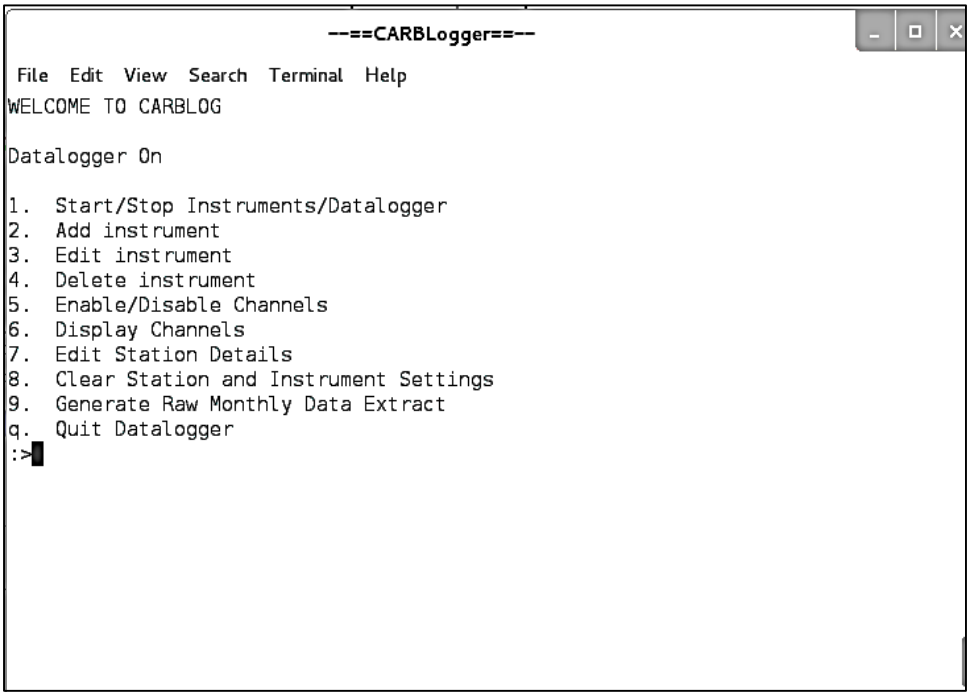


Figure 3.1: CARBLogger Home Screen.

To enter/proceed with each submenu, from the main menu, enter the number of the submenu you wish to access, followed by pressing the "ENTER" key, which will bring up the submenu.

3.3 Start/Stop Instruments/Datalogger:

The Start/Stop Instruments/Datalogger menu (Figure 3.2) allows the operator to start or stop a single, multiple, or all instrument drivers from running on CARBLogger. Stopping an instrument here prevents data from being collected by CARBLogger. This is used when an instrument is removed or stopped for an extended period of time.

In contrast, when disabling a channel the CARBLogger will continue to collect data for DMS, but flags each data point with a user selectable meta-tag instead. The instrument could be under maintenance or calibration for a short period of time.

For details of Enable/Disable Channels, please refer to section 3.7 of this document.

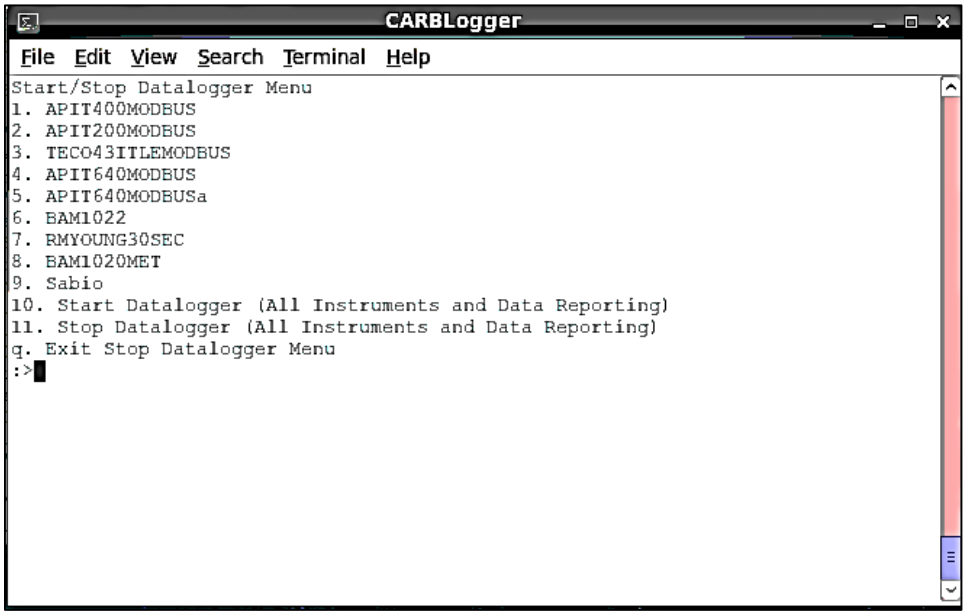
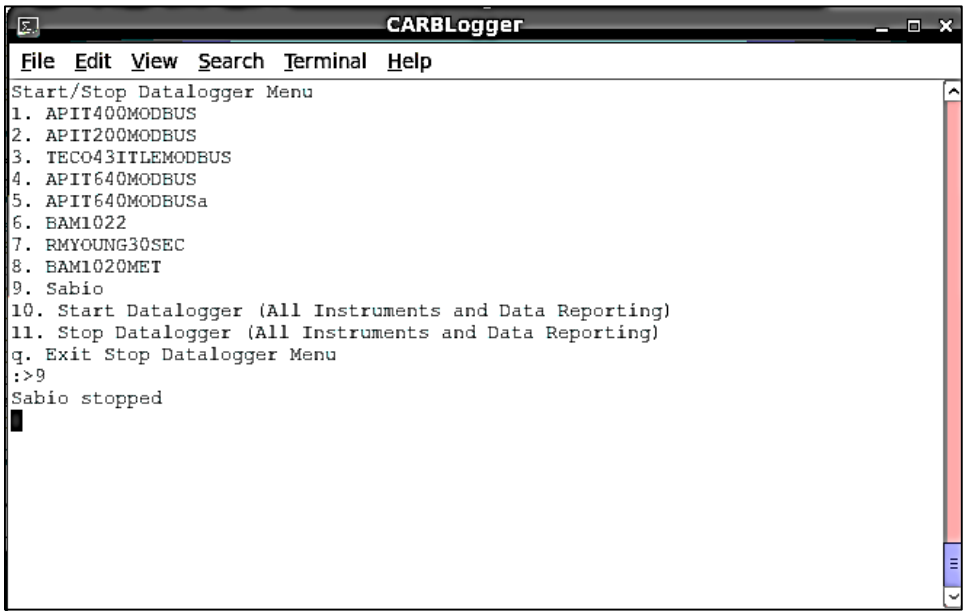


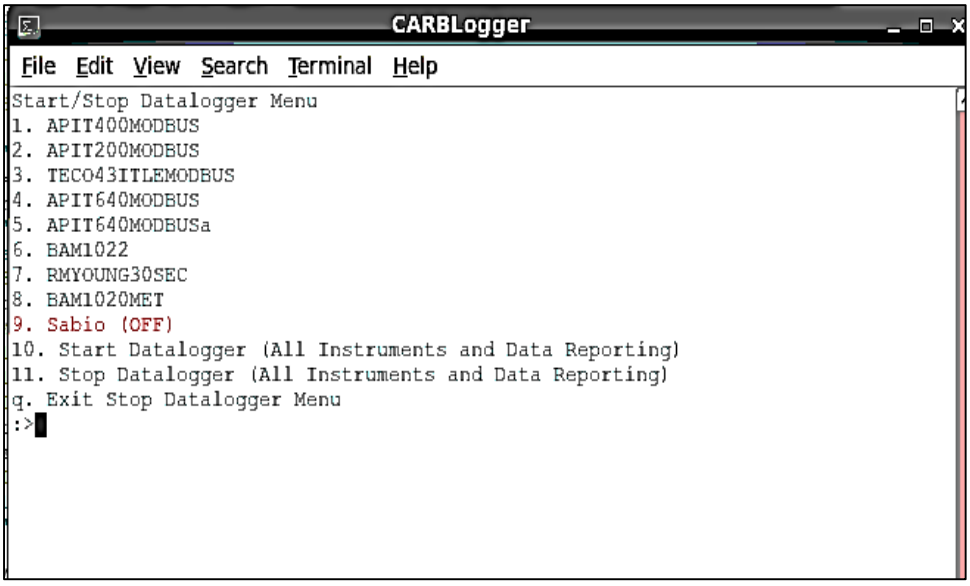
Figure 3.2: Typical Start/Stop Instrument Datalogger Menu.

To stop an instrument, for instance, the Sabio instrument driver, choose the number displayed in front of the Sabio under this menu, in this case the #9, and then press enter to proceed. Refer to Figure 3.3a.

The message "Sabio stopped" should display, followed by the same menu, with Sabio displayed in red and the message (OFF) next to it. Refer to Figure 3.3b.



a) Stopping the Sabio unit by selecting #9



b) Sabio is stopped with (OFF) displayed in red

Figure 3.3: Screenshots on How to Stop an Instrument Driver.

Finally, choosing the last item in this menu "Stop Datalogger (All Instruments and Data Reporting)", will stop all instrument drivers from collecting data, as well as stop all data ingestion and reporting processes. This option is rarely used but may be needed when the entire monitoring site is undergoing maintenance or any unexpected circumstances.

3.4 Add Instrument:

The "Add Instrument" screen (see Figure 3.4) allows operators to choose the instrument driver to be added to the existing CARBLogger configuration. Only drivers that are made available on that specific CARBLogger box will be displayed and selectable on the list.

The order and number assigned to the displayed drivers depend on the state of the CARBLogger program, so users should pay attention to the menu screen. The following conditions should be kept in mind when using the interface to add new drivers to CARBLogger:

- At the time of this writing, it is not advised to attempt and add a second occurrence of the same driver/instrument. Although the interface may appear as if you have added a second instrument, it will not work properly.

- Collocation of an identical instrument should be addressed by choosing (or

requesting) the "_a" variation of the same driver. For instance, in the above screen APIT640MODBUS is the primary driver and APIT640MODBUS_a is the co-located version. Remember that the dms.cfg file must be properly configured manually in order to report primary and co-located data to DMS correctly.

- Only one calibration source can be handled for DMS reporting at a time. Choosing two or more "calibrating" drivers such as LGR_N2O_CO_drv, APIT400IZS_drv, environics81000_drv, and the Sabio4010_drv will cause anomalous data flagging and reporting, although the raw data will still be collected by CARBLogger.

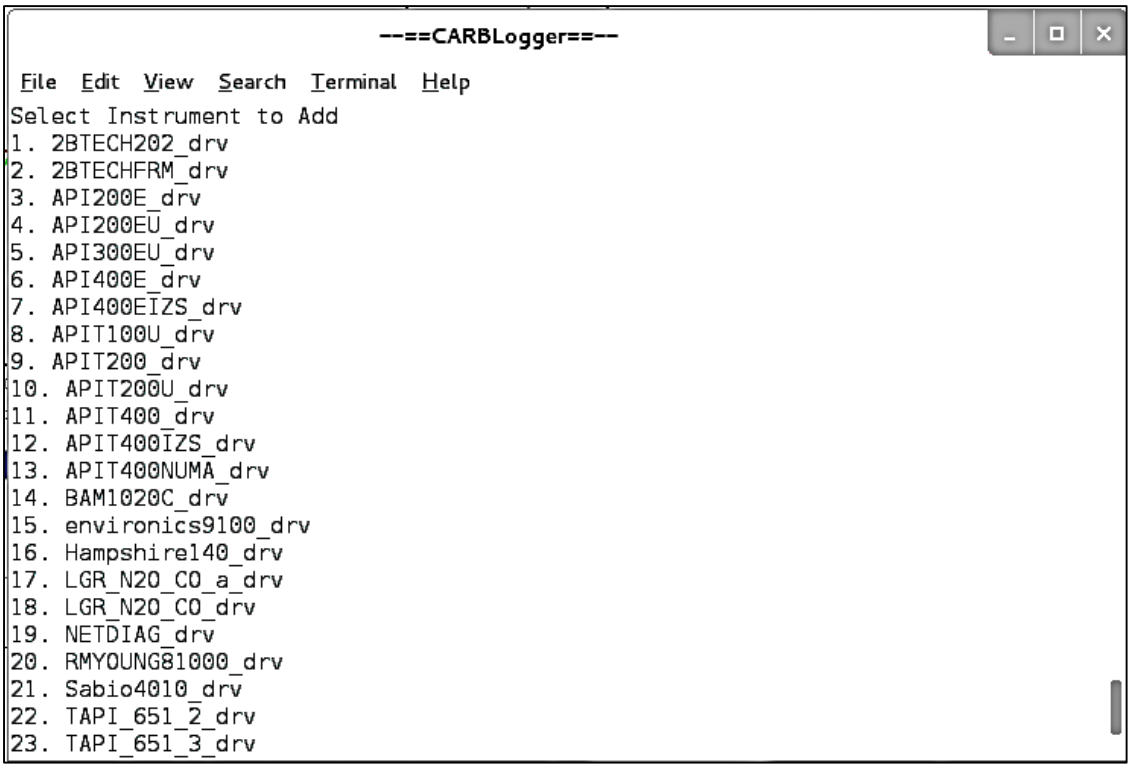


Figure 3.4: Typical Add Instrument Menu

To add an instrument driver to the system, choose the instrument by typing the number listed on this menu, adjacent to the desired driver, followed by pressing the enter key. The prompt will then take you to the "Edit Instrument" screen for the instrument you are currently adding. For details of the Edit Instrument menu, please refer to next section 3.5.

To cancel the request, pressing the "q" button followed by the enter key, will return you back to the main menu.

Note: For additional details on adding an instrument (driver) to CARBLogger, please refer to Section 4.4 of this document.

3.5 Edit Instrument:

If not accessed via the instrument addition menu, the first screen (see Figure 3.5) presents a selection of currently active instruments you can choose to edit:

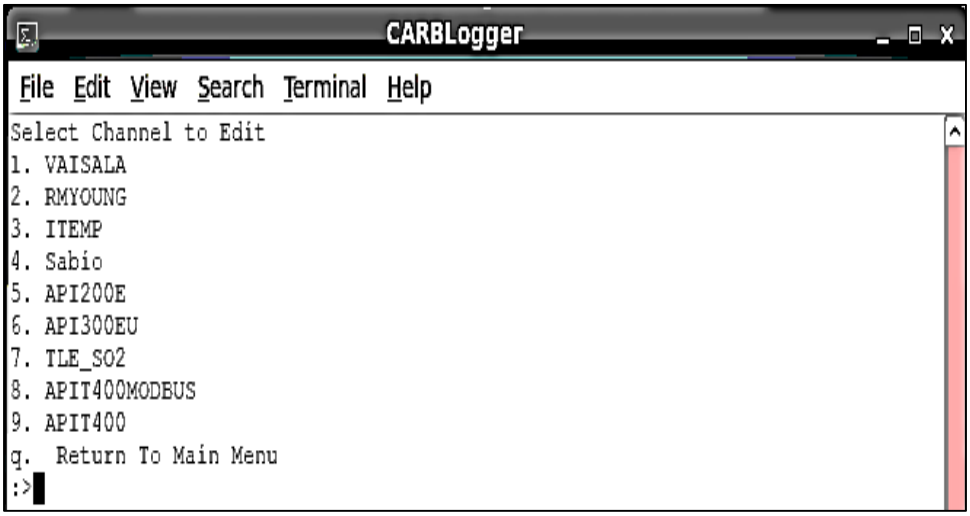


Figure 3.5: Typical Edit Instrument Menu

Choose the instrument you wish to edit by typing the number adjacent to the instrument, and then press enter. For example, here we will edit the APIT400 driver by pressing #9 and then the Enter key.

Different drivers will have different parameters to change, depending on the make, model, firmware version of the instrument, as well as the manner in which the driver has been written. Some common options are shown below (Figure 3.6).

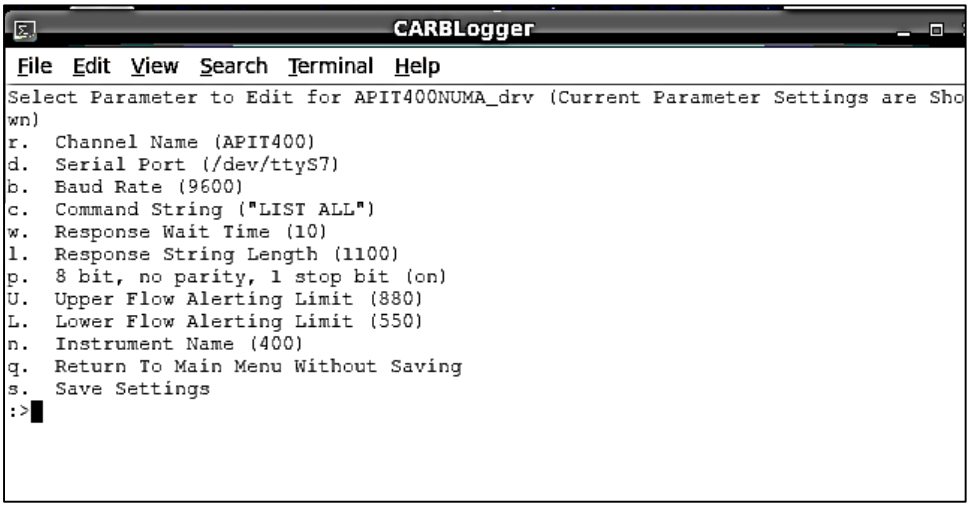
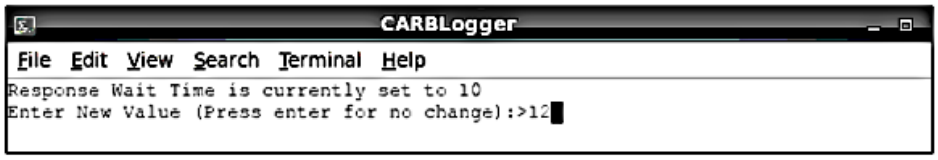


Figure 3.6: Common Editable Parameters from an Instrument Driver

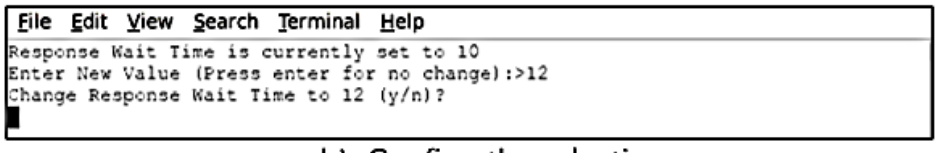
Pressing the case sensitive letter adjacent to the parameter you wish to change, followed by the Enter key will allow you to futher edit that parameter.

As instruments and communication boards age, we may need to alter the wait time needed to obtain a response from the instrument. To update the intrument's response wait time, select that option and then press enter.

For instance, change the response time from 10 to 12, by pressing "w" followed by the Enter key. After pressing enter, the menu will prompt for confirmation. Press "y" to confirm your choice. See figure 3.7 below.



a) Update the new value



b) Confirm the selection

Figure 3.7: Screenshots: How to Edit and Instrument Driver.

Pressing the "y" key followed by the Enter key returns you to the parameter's edit screen. Finish making any additional changes, then press the "s" key to

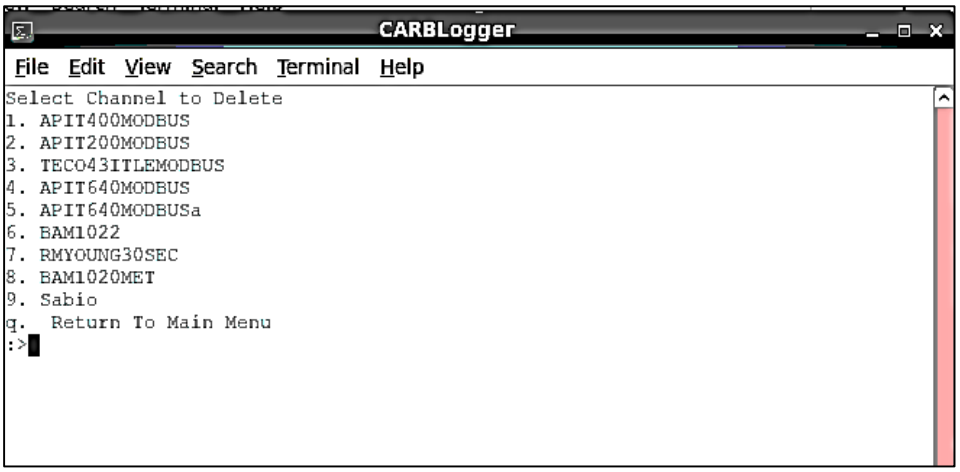
save the setting changes. The prompt will return you back to the instrument selection menu of the Edit Instrument submenu. From there press the "q" key to return to the main menu. All drivers will stop and restart.

The changes made to the parameters setting for each instrument driver have been altered by the interface, so when each driver restarts it will run using the new settings.

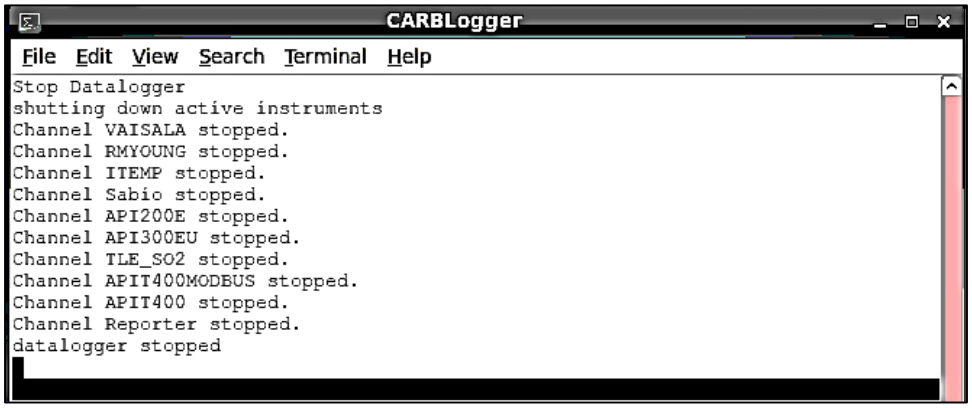
3.6 Delete Instrument:

When an instrument will no longer be collecting data on a CARBLogger, the Delete Instrument menu can be used to delete the instrument currently being run. Deleting the instrument means that the CARBLogger interface will no longer store the configuration of the instrument, and that driver will not be automatically started during reboot. Also raw and diagnostic data will stop reporting.

To delete an instrument, go to the Delete Instrument menu (see Figure 3.8), select the existing instrument that you want to delete from the system, enter the instrument number, and then press the "y" key to confirm. Once the deletion is complete, press the "q" key to return to the main menu; CARBLogger will reboot.



a) The List of Instruments Available from the Delete Instrument Menu



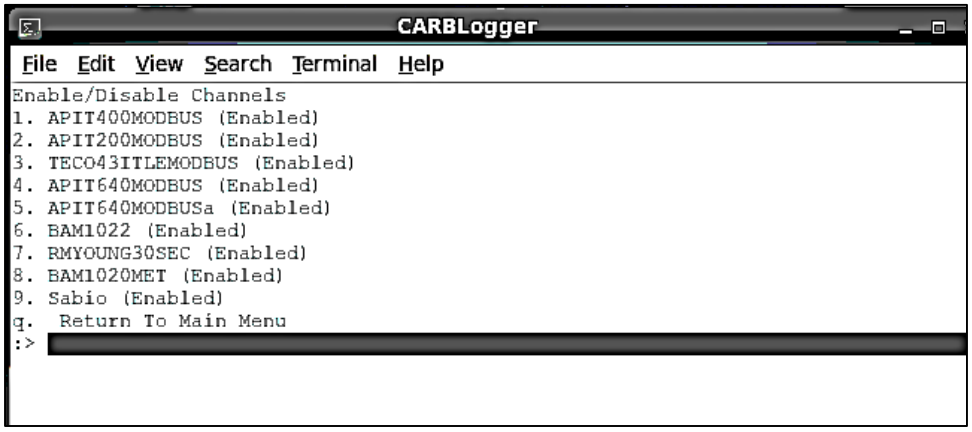
b) CARBLogger Reboot After Deleting an Instrument Driver

Figure 3.8: Typical Delete Instrument Menu

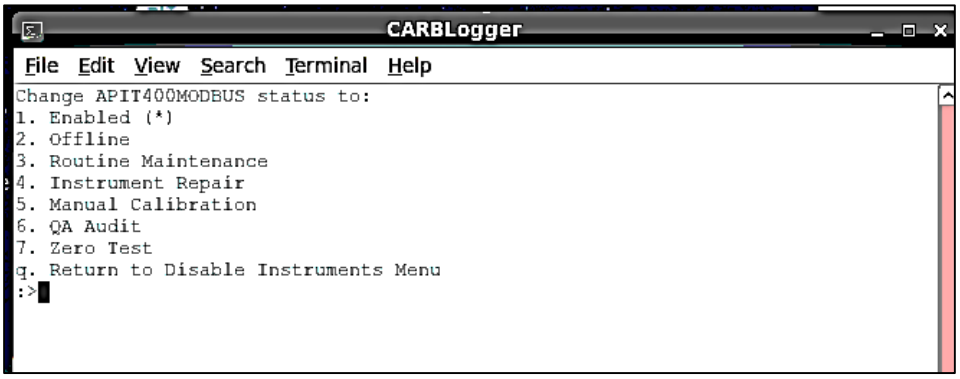
If an instrument is deleted and needs to be reinstated, it will need to be added back to the system using the Add Instrument menu.

3.7 Enable/Disable Channels:

When site maintenance or other activities occur which compromise the quality of data collected by any single instrument, it is advisable to use the Enable/Disable Channels menu in order to set the correct operational code to the data produced by the affected instrument(s). Doing this would allow DMS to infer should the data point be assigned an invalid QC code.



a) List of Channels Available from the Enable/Disable Channels Menu



b) List of Operational Statuses Available for Selection

Figure 3.9: Typical Enable/Disable Channels Menu.

Follow the similar steps that have been described in the previous section for instrument/channel selection. CARBLogger will not reboot from a change of operational status for an instrument.

Since DMS calculates the hourly average by averaging all valid minute values during an hour, **a failure to properly disable instruments when performing maintenance, calibrations or other activites** can result in false values being calculated by DMS and reported to real time data clients.

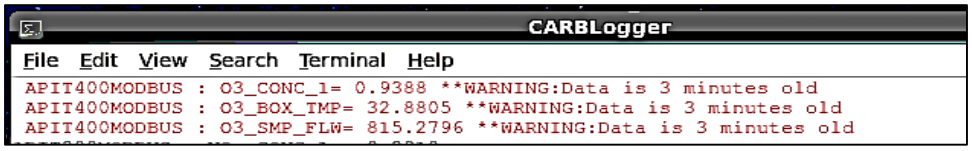
3.8 Display Channels:

The Display Channels screen from the CARBLogger interface is useful for viewing the real time status of raw data being collected from instruments. The number of displayed parameters is configured through instrument drivers. The parameters will be displayed in a line-by-line format, and the data being displayed on the Display Channels screen will be updated every 15 seconds. In addition to the raw data value being collected, this screen also offers several pieces of information regarding the state of data collection.

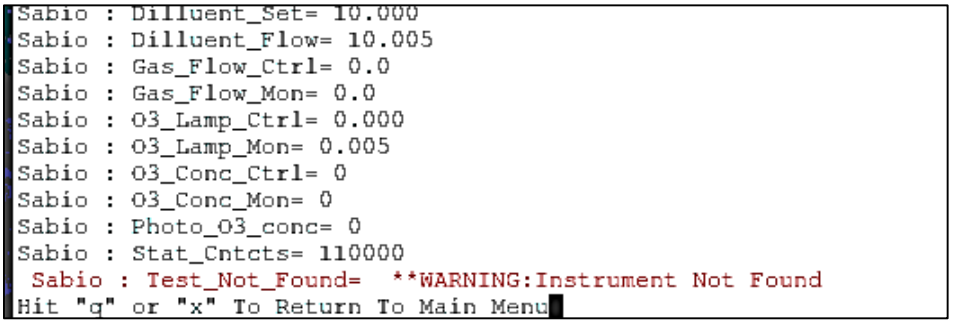
If an instrument or any single parameter has not reported for more than one of its regular reporting intervals, text appears in red saying the amount of time since the last raw data point was recorded will begin to appear. See Figure 3.10a.

If an instrument has not reported data for more than a day (24 hours), red text stating “Warning: Instrument Not Found” will be displayed. See Figure 3.10b.

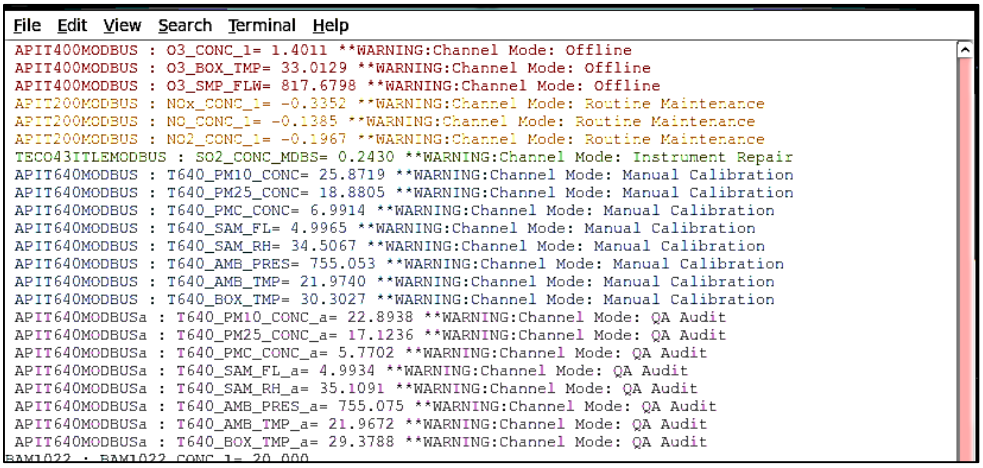
Finally, if an instrument is flagged offline for any reason, a verbal description for each type of offline status will be displayed. See Figure 3.10c. The colors of the alert are randomly assigned to differentiate the instruments installed.



a) Warning message for non-current values



b) Warning message for inactive parameter



c) Warning messages for different offline statuses

Figure 3.10: Different Error Message from Display Channel Screen.

3.9 Edit/Clear Station Details:

The Edit Station Details screen (See Figure 3.11) is used to modify station specific parameters that affect all instruments, including the station code, site elevation, the two diagnostics reporting hours, and the email contacts for reporting, etc.

The Clear Station and Instrument Settings screen (see Figure 3.12) is used to reset all station specific parameters and all instrument settings to the default blank state.

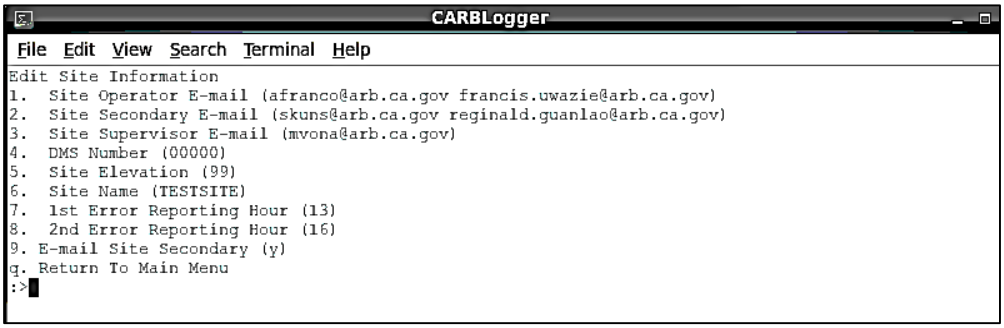


Figure 3.11: Edit Station Details Screen.

Note: Station details should be the first priority for configuration during any initial CARBLogger setup. For details of site configuration, refer to Section 4.1.

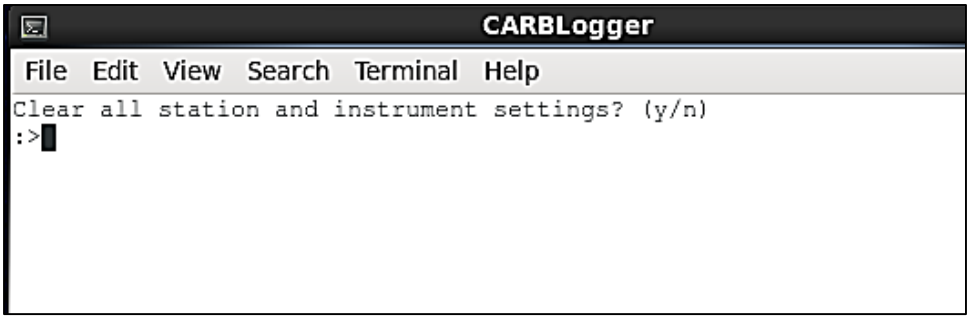


Figure 3.12: Clear Station and Instrument Settings Screen.

Note: To clear station and instrument settings from CARBLogger, select the specific option from the main menu. A new prompt would appear to ask for confirmation. Enter "y" to confirm. Once executed CARBLogger and any existing station and instrument settings will be reset to the default blank state.

3.10 Generate Raw Monthly Data Extract:

A user may want to view the raw data for one instrument in particular and view it on the work station. In such a case, this menu option can be used to scrape the raw data, extract the parameters requested and send the data to the provided email address. See Figure 3.13.

Go through the requested field one by one, and the CL prompt will direct users for the proper values to enter. Select Generate Report for the data extract file.

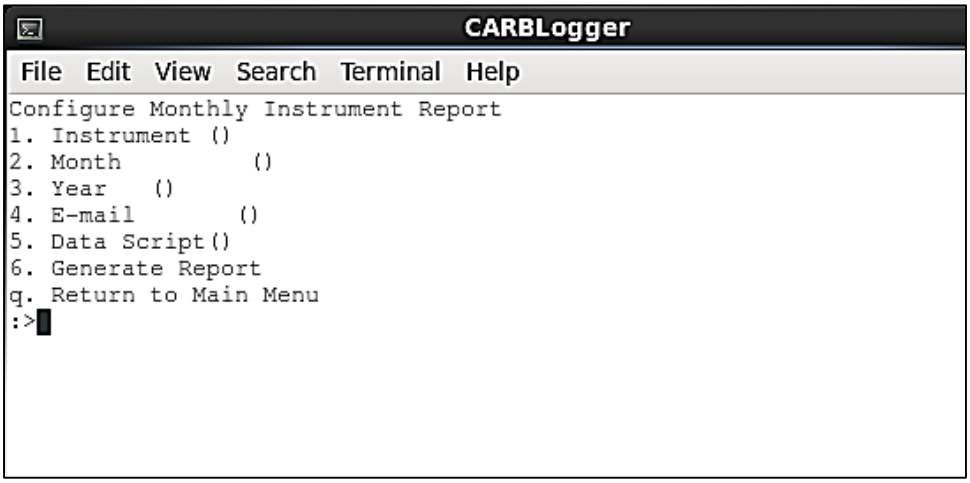


Figure 3.13: Generate Raw Monthly Data Extract Screen.

Note: The extracted raw data from this menu is in its own format and is not appropriate to use for backfilling data gaps on DMS. To backfill data gaps, please follow the procedures from Section 7.0 of this SOP document.

4.0 CARBLOGGER CONFIGURATION

4.1 Initial Station Configuration:

Prior to using CARBLogger, a site's parameters must be properly configured. Most site parameters used by CARBLogger can be found in the Add Instrument and Edit Station Details screens (as shown in sections 3.4 and 3.9 respectively). In this section, details will be provided for each required setting.

Note: Not ALL of the site parameters are editable directly on the screen. If needed, changes can be made by editing the "~/CARBLog/Config/config" file via the Terminal Shell. For instance, manual editing with this config file is needed, when switching serial card from Perle to StarTech, when adding new Modbus drivers, and when switching calibrator between Sabio 4010 and Environics 9100, etc.

The following table includes the description and location of each parameter requiring configuration prior to using CARBLogger.

Table 2: Description of Required Parameters:

Parameter	Location	Description
Site Operator, Supervisor, and Secondary Email Addresses	Edit Station Details (Menu)	Set the email addresses which receive emails from this CARBLogger. *Note: Multiple email addresses can be used by providing a space separated list for any one value.*
DMS Number	Edit Station Details (Menu)	This is the 5-digit CARB identification number which is appended to every data record produced in CARBLogger. If this number is wrong, it is possible to contaminate other site data in DMS (if DMS site verification has been disabled for the site).
Site Elevation	Edit Station Details (Menu)	Various data clients including Air Now and AQS require site elevations. Some instrument diagnostic values such as flow also require site elevation. To date, no drivers or DMS import processes use this field.

Parameter	Location	Description
Site Name	Edit Station Details (Menu)	This is the name of the station used in email notifications from the CARBLogger, and it only appears in the emails sent to station operators.
First and Second Error Reporting Hour(s)	Edit Station Details (Menu)	The two reporting hours when the summary of site operational error conditions are transmitted. If set to 05, the first email will be sent at 0500 hours in using the GMT - 8 time zone.
E-mail Site Secondary	Edit Station Details (Menu)	This value can be set to Y or N. If set to Y, the site secondary (or calibrator) will receive copies of the error emails.
Instrument Drivers	Add Instruments (Menu)	One driver should be added per each instrument. In order to do this effectively, determine the physical port connections for each instrument which can vary depending on the make and model of the serial card, or if MODBUS drivers are used or not, etc.
DMS Data Reporting	The dms.cfg file (Terminal Shell)	For every data label displayed on the CL screen, each label in the dms.cfg file must map to the appropriate DMS counterpart, as shown in DMS on the Parameters screen.

4.2 Configure Station Details:

To properly configure a site's station details in CARBLogger the operator needs to enter the Edit Station Details screen (by typing "7" followed by the Enter key from the CARBLogger main menu).

From the Edit Station Details menu, the operator is presented with a list of options to update. It is critical that all these items be updated. Simply choose each of the items presented and follow the on screen prompts.

Shown below (figure 4.1) is an example of the station details configuration for the Test Site (00000) located in the ODSS Instrument Shop.

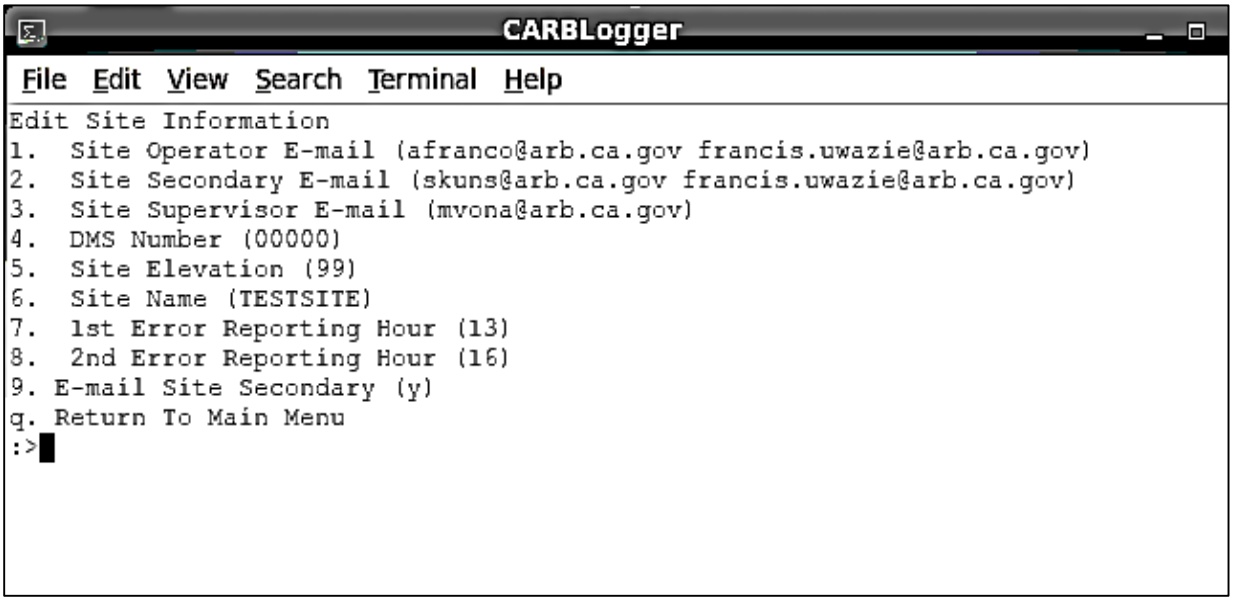


Figure 4.1: Example of Station Details Configuration.

Note: The DMS Number field is the station ID assigned in DMS for each monitoring site. This ID is synonymous with the CARB Site Number. CARBLogger uses this number to identify itself when transmitting data to DMS. See Appendix G for a list of the five-digit site numbers for all CARB Air Monitoring Stations which were in operation at the time of this writing.

4.3 Instrument Configuration Settings:

Once CARBLogger site information has been configured, the logger must be configured to know which instruments are operating at the station, and on which physical port the instrument is located.

Prior to adding any new instrument drivers, the operator must confirm that each instrument's communication settings are known, as these will be requested by the CARBLogger prompt for each instrument's setup dialog.

Every setting should be verified for the instrument prior to adding it to CARBLogger. The following table (Table 4.2) directs users how to verify instrument communication settings, depending on the type of connection protocol the instrument uses for connection.

Table 3: Different Ways to Verify Instrument's Communications Settings

Protocol Type	Specificity	Procedure
Serial	Chassis Serial Port	See Appendix C.
Serial	Perle Serial Card	See Appendix C.
Serial	StarTech Serial Card	See Appendix C.
Ethernet	MODBUS Over TCP/IP	This only works on CENTOS or other Linux distributions with the modpoll module installed. Open a terminal, and type the following command: *modpoll -m tcp -p 502 -1 -t 3:float -c 2 -f xxx.xxx.x.x* where "xxx.xxx.x.x" is the assigned IP address for the instrument. Examples: API200 = 172.16.0.2, API300 = 172.16.0.3, API400 = 172.16.0.4, and Teco 43i = 172.16.0.10 It is also possible to connect an instrument to the DHCP provisioning switch, however, this is not advised. If the IP address, the CARBLogger, the intermediate switch, and the instrument communication settings have been correctly configured, the following message should display: modpoll 3.4 - FieldTalk(tm) Modbus(R) Master Simulator Copyright (c) 2002-2013 proconX Pty Ltd Visit http://www.modbusdriver.com for Modbus libraries and tools. Protocol configuration: MODBUS/TCP Slave configuration...: address = 1, start reference = 1, count = 2 Communication.....: 172.16.0.4, port 502, t/o 1.00 s, poll rate 1000 ms Data type.....: 32-bit float, input register table Word swapping.....: Slave configured as big-endian float machine
Ethernet	Windows File	PICARRO G1301 uses file sharing. Open the

Protocol Type	Specificity	Procedure
	Sharing	gigolo file browser, confirm that you can browse to the file system on the remote instrument.

It is very important that the instrument configuration settings are entered and updated correctly. If not set correctly, reported data could become incomprehensible and invalid for submission. Table 4.3 illustrates some common instrument configuration mistakes.

Table 4: Examples of Incorrect Instrument Configuration Changes

Data Stream Before Change	Data Stream After Change	Result
O3= 4.5 ppb	O3= 0.005 ppm	If the driver assumes that the ozone value is being reported in parts per billion, DMS may begin to receive artificially low numbers because the driver assumes results are reported in parts per billion.
O3= 4.5 ppb	Ozone= 4.5 ppb	If the driver is looking for the data marker "O3=" and instead is presented with "Ozone=", the ozone data stream will stop completely.
4.5 80 33	4.5 33 80	Some instruments do not mark their data, but rather return their data in a single ordered string of values. In these situations, a principle sometimes referred to as "token counting" is used to make the drivers. A change in the order of the data will therefore cause the wrong numbers to be reported to DMS.

Many of these mistakes can be mitigated by reviewing the raw data files and adjusting the parameter configuration in the related instrument drivers. Instrument operator should notify CARBDMS of the errors, and correction/adjustment to the drivers shall be reviewed and made by CARBDMS staff only, unless other instructions are given. The corrected data shall be re-ingested into CARBLogger and DMS.

At the present time, only one calibration system (i.e. Environics 9100, Sabio 4010 or API IZS) can be used per instance of CARBLogger to properly configure the dmsout.sh process flag calibration, flagging, and formatting for DMS (if required). Some instruments (API400E IZS) have their own internal calibrator, they will record their own 'virtual' calibration records into its raw data record. If a Sabio 4010 was to run on the same CARBLogger as an API400E IZS, the calibration reporting paradigm used to flag data for DMS would be incorrect 50% of the time.

4.4 Adding Instrument (Driver) to CARBLogger:

This section will cover additional details on adding instrument (Driver) to the CARBLogger. For basic description of the Add Instrument menu, please refer to Section 3.4 of this document.

After entering the Add Instrument menu, the operator is presented with a list of drivers available to CARBLogger (See figure below). This list is typically populated by the drivers available on the specific CARBLogger distribution.

Note: If you do not see or recognize the driver needed for your particular instrument, it is possible that a driver for that instrument has not been created or added to your particular CARBLogger. If this is the case, contact the CARBDMS Team for assistance.

Note: CARBLogger should update its driver's inventory automatically through syncing to the CARBLogger repository in SourceForge.net. This "push" of CARBLogger driver updates allows CARBLogger to remain updated for glitch/bug fixes, driver modifications or new driver releases. Automated CARBLogger updates currently propagate to the entire network on a weekly basis.

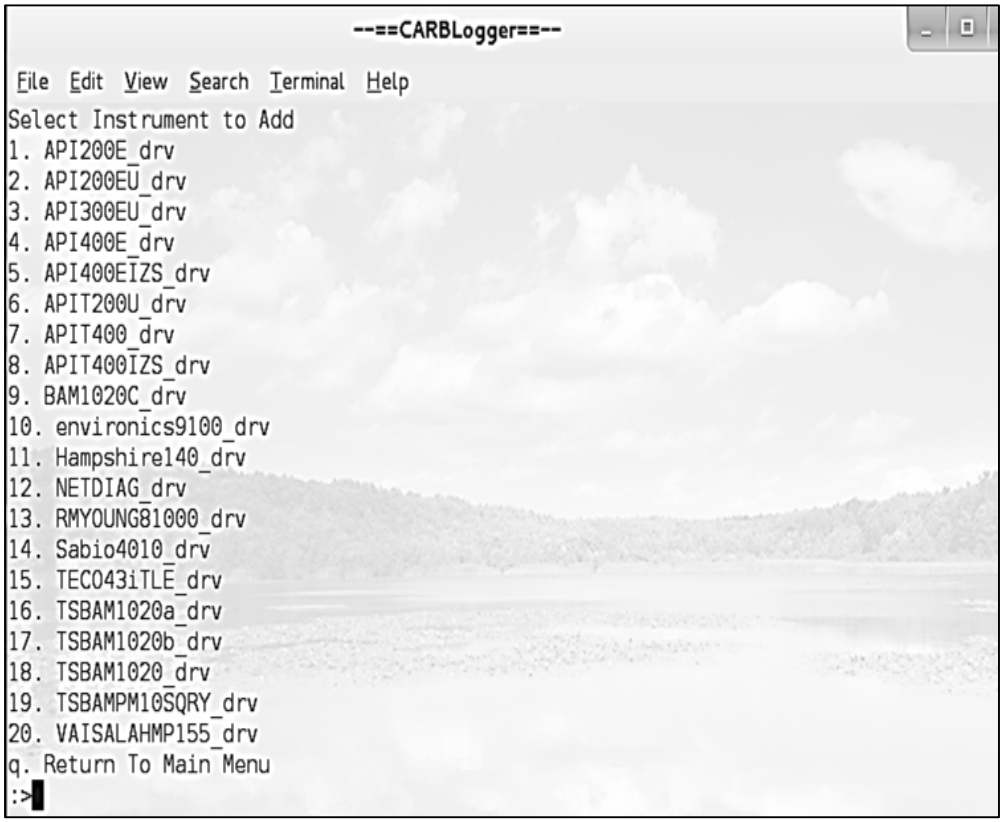


Figure 4.2: List of available Instrument Drivers.

In general, CARBLogger drivers will be named after the make and model of the instruments. For example, to begin logging data with an APIT400 using IZS function, its driver will need to be added to the CARBLogger. From the above menu shown, one would type in "8" and then press enter to select the APIT400IZS driver.

In some cases, network instruments may have been purchased or implemented with particular hardware options which require new or different data processing. In those cases, these specific conditions or qualifiers are incorporated into the driver name. The following table lists those drivers which should be used when special configurations of instruments have been made as of version 1.2.

Table 5: Drivers of Special Configuration

Driver Name	Implementation Specifics
API400EIZS_drv, or APIT400IZS_drv	For monitoring sites without any gas calibrator, analyzer with an internal zero-span (IZS) option, e.g. API400EIZS or APIT400IZS, has been utilized. When an IZS analyzer is used, special driver is needed to query the state of the internal IZS unit in order to generate correct calibration channel information. For this reason, an IZS version of the driver has been written, and has been indicated by appending "IZS" to the model name in each case. Although the non-IZS drivers would still collect data from both instruments, the information needed for CARBLogger to pass calibration flagging information back to DMS will not be present in the raw data stream. The IZS drivers will automatically configure the sys_resources/dms.cfg file. NOTE: Do not implement either of these drivers with any calibrator or with any other instrument driver which contain their own internal calibration channel.
BAM1020C_drv	At the time of the driver's writing, the "C" was used to indicate PM Coarse, which is essentially (PM10 – PM2.5) collected from tandem PM10/PM2.5 BAM1020 instruments connected by a serial connection and running special firmware. At the time of this writing, only the Fresno air monitoring station runs this type of unit.

Driver Name	Implementation Specifics
TSBAM1020_drv, TSBAM1020a_drv, or TSBAM1020b_drv	The TSBAM1020 driver is named to indicate that this is the "Time Setting" version of the BAM1020 driver. There are no longer any production BAM1020s which would still use the non-time setting variant of this driver, since all analog time setting mechanisms should have been removed from the newer BAM1020 models. The primary BAM located at any production air monitoring station should use TSBAM1020_drv. The difference between FEM, non-FEM, and collocation is accounted for in the DMS configuration section.
TSBAMPM10SQRY_drv	The U.S.EPA determined that AQSB should report PM10 in Standard Condition, while other data clients require PM10 in local conditions. This driver was written specifically to work with the "Query Port" option/implementation of the BAM1020 while running the PM10 hat. This driver requires two serial connections to the time setting BAM, one for the standard serial port and one for the query port. NOTE: Unless this BAM is using the PM10 hat and running the query port, the TSBAM1020 driver should be used with modifications, as will be discussed in the DMS configuration section.
Sabio4010D_drv	Unlike the Environics 9100 driver, the Sabio4010D has user configurable equilibration and recovery time settings. Using this driver requires a different version of the CARBLogger database, stored procedures, and a specific configuration of the DMS server for the site.

5.0 DATA FLAGGING AND OPERATION

CARBLogger is an evolving, open-source project. Changes introduced by instrument manufacturers, software applications, procurement and information technology policies necessitate constant modifications to the underlying processes employed by the logger. We will discuss the operation of only the latest, most common version of CARBLogger here, and the guidance in this document may change as this project evolves.

Note: CARBLogger will remain operative and continue to collect data even if the station's internet connection is down. Once internet connection resumes, CL will push data collected during connection outage to DMS.

5.1 Data Flagging Introduction:

Operational codes (Op codes) are DMS data flags assigned to every data point that CARBLogger sends to DMS. "Data flagging" is the process of altering these Op codes in order to communicate data quality and purpose of every value. Without data flagging, DMS is unable to automatically calculate valid hourly averages, calibration values, or assign null codes. CARBLogger automates data flagging in two manners, both requiring input from the station operator.

Manual data flagging –

Operators have the ability to manually mark down instruments using the CARBLogger interface when they need to perform a physical process on the instrument. Common processes include calibrations, audits, maintenance, or zero testing (See section 3.7, and Appendix I). Manual operational codes have a higher priority than those assigned by automatic processes. Currently, CARBLogger has the ability to assign 7 different manual Op codes to the data it collects.

Calibration Data Flagging –

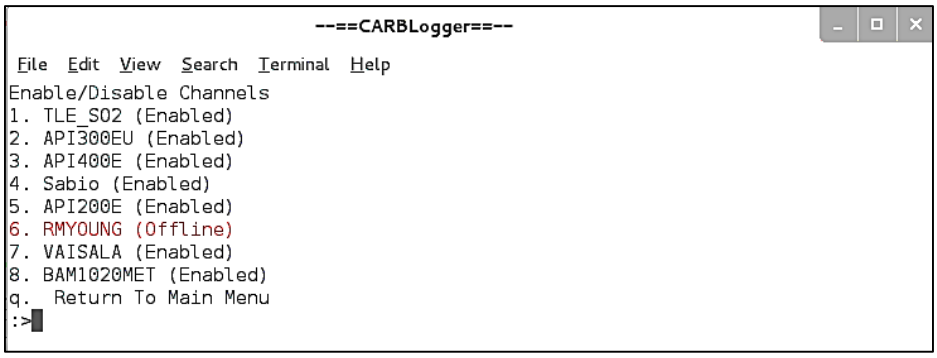
Station operators must configure the dms.cfg file to express how their station is physically plumbed. Automated gas calibrators and some self-calibrating instruments produce a calibration data stream. Using these two pieces of information, CARBLogger interprets calibrator effects on the quality of the instrument data, and assigns each data point the appropriate Op code.

Note: At the time of this writing there can be only one calibration data stream.

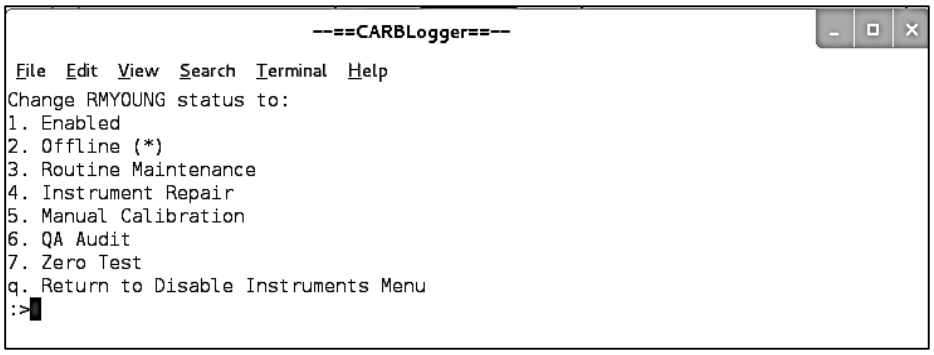
5.2 Manual Data Flagging:

Often a site operator may find it necessary to take an instrument off-line (i.e. require the instrument to collect data), but prevent DMS from averaging that data into valid ambient data. This is done by Manual Data Flagging.

Manually flagging down a channel uses the Enable/Disable Channels menu from the CARBLogger interface. CARBLogger will continue to collect data from the instrument, but data will be flagged if reported. After data is sent to DMS, the Op code will be used to omit data from averaging into valid data values.



a) Enable/Disable Channels Menu



c) Selecting an offline status for the disabled channel

Figure 5.1: Marking Down a Channel/Instrument

Selecting any instrument from the above menu will allow you to assign any non-valid Op code (as described in Appendix I) to the next data point collected from that instrument.

DMS will map a QC code for the hour, based on the average of the highest QC codes from the 60-minute period representing that hour. The QC codes are assigned by DMS based on the Op code assigned by CARBLogger.

If an operator marks an instrument as "offline" in CARBLogger, each

subsequent minute will receive a "9" Op code. Each similarly coded minute value will receive a "9" QC code from DMS. If this happens for more than 15 minutes in an hour, that hour will be QC coded as invalid by DMS.

This point becomes especially significant when flagging instruments which collect data points using intervals greater than one minute.

The Met One BAM1020 driver collects a single data point once per hour. This driver has a variable which allows you to set the two-digit minute at which the data logger queries the instrument. Assume that the variable is set to minute 04 (the default value).

The state of the Op code is set at the time that the data point is collected, not the time that data point is measured. In the case of the BAM1020, the last column of the dms.cfg file has a value of 60 for the offset which means that each data point it collects represents a data point 60 minutes earlier.

This is also significant with other instruments with larger data query interval (i.e., 15 minute intervals), such as the Magee Scientific Aethalometer.

5.3 Calibration Data Flagging:

To properly flag data for DMS, a reporting process referred to as dmsout.sh is used to produce flagged data by comparing the raw text data from instruments against a site's calibrator output. CARBLogger data flagging schema is configured using the "dms.cfg" file.

Properly configuring the dms.cfg file not only determines which data will be reported to DMS, but also flags data based on the manifold which the instrument is connected, the state of the calibrator, translates the CL data name to the proper DMS parameter label, and time stamps "raw data" stream using Network Time Protocol (NTP). This timestamp corresponds to the time at which the data point was acquired and ensures data collected from instruments are time synchronized with CL.

Figure 5.1 displays a Venn diagram which illustrates the relationship of data collected by and reported from CARBLogger. CARBLogger displayed data is a small subset of data that was first collected in raw text format. The nightly data extracts and data displays are both produced using this data subset. Of this displayed data, only the data with data labels added by each driver and configured in the dms.cfg file will be submitted to DMS. Finally, only a portion of this DMS data is reported to the DMS data clients.

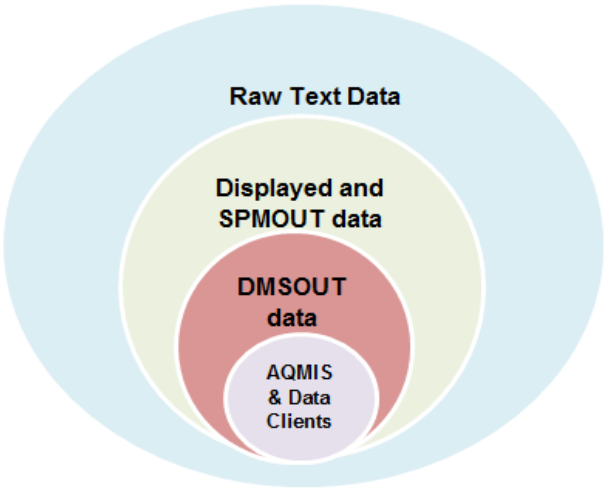


Figure 5.2: Venn Diagram of data collected by CARBLogger.

To auto report data to DMS, the following conditions must be met (as this requirement is enforced by AQSB for system integrity):

- The driver which corresponds to the instrument collecting the data must be properly configured and running.
- When CARBLogger is running, the required data from each instrument data channel must have been displayed on the Display Channels screen.
- If flagged data is required, either the calibrator driver must be running, or the instrument's own driver must provide calibration capabilities.
- The reported data channel must have at least one data label entered in the "dms.cfg" file.

5.4 DMS Configuration File:

Note: ODSS typically provides full support for this portion of the CARBLogger implementation, so an in-depth understanding by the user will not be required but will be documented here.

The DMS configuration file (dms.cfg) is basically a space separated set of rules which describes how each instrument is interconnected. The manifold to which the instrument is connected, the sampling latency of an instrument, and the way that DMS has been configured to accept data are all expressed in this configuration file.

For instance, during a calibration sequence, for CARBLogger to know that the SO2 parameter must be flagged invalid while an ozone calibration cycle is running, CL must know that the SO2 instrument is connected to the same manifold, and it must also know that the running calibration phase is unrelated

to any maintenance routines used by the SO2 instrument.

Although the dms.cfg file is space separated in CARBLogger, we will use a table here to discuss the various values and the impact they will have on the data being reported to DMS. We will also add a Line Number column here to allow for ease of reference.

Note: If any line of the dms.cfg file has a pound (#) sign in front of it, it means that line will be ignored from execution.

Table 6: Tabular View of the DMS.CFG Settings.

Line Number	Data Label	DMS Label	Zero	Span	Prec.	Minute Offset	Manifold System
1	NOX=	NOx	11,21	12	22	0	1
2	NO2=	NO2	11,21	13	23	0	1
3	NO=	NO	11,21	12	22	0	1
4	O3=	O3	11,21	14	24	0	1
5 (Environics)	RECOV=	RECOV	15,25	--	--	0	1
6	5020_SO4=	SO4	31	--	32	15	3
7	BAM1=	BAM10	--	--	--	60	9
8	BAM2=	BAM25	--	--	--	60	9
9	BAM3=	BAM25 a	--	--	--	60	9
10 (Sabio Only)	RECOV=	RECOV	66	--	--	0	1

To illustrate how this works altogether, consider a single morning calibration for an ozone instrument, a NOX instrument, and a BAM1020. We will show the collective calibration, and then dissect each calibration for each instrument in terms of the portions of the dms.cfg file that affect it. Refer to Figure 5.3 below.

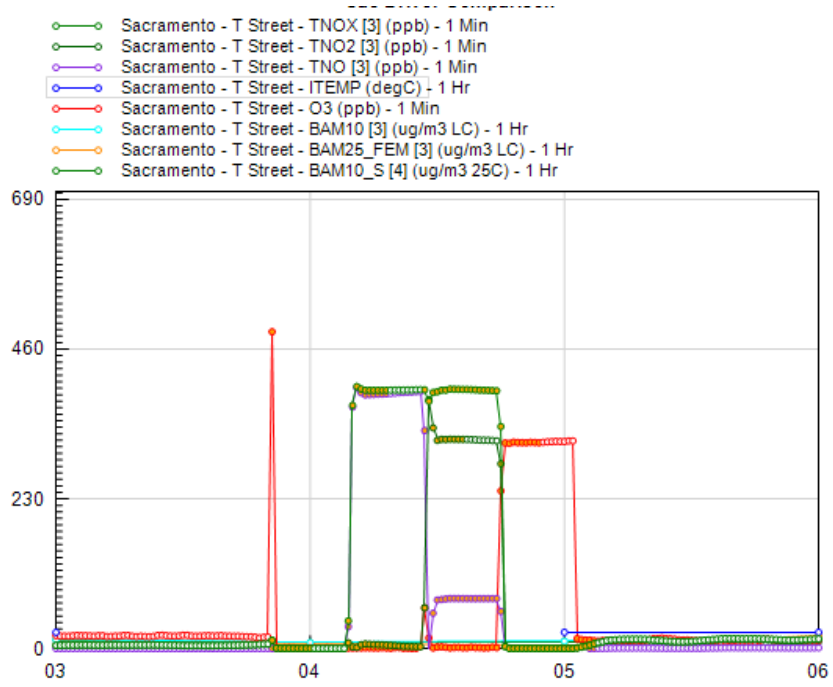


Figure 5.3: Sample Calibration Data Capture from DMS.

The line entered into the dms.cfg file for the ozone channel for the above graph reads as follows:

O3= O3 11,21 14 24 0 1

- The first value/column "O3=" corresponds to the value on the display of the CARBLogger. In the raw data record, this tag will appear immediately before the value (for instance O3=4.231). This data tag is added by each driver.

- The second value "O3" corresponds to how the DMS administrator has configured the system to accept the ozone values for your station. In this case, DMS accepts O3, and in other cases (i.e., in the event of collocation) the administrator may ask the data to be reported with an affixed "_a" in which case the second value would become "O3_a".

- The third value/column represents a comma separated list of sequence and step combinations which should be recognized as "zero" gas. In this case, sequence 1 step 1, and sequence 2 step 1 are both zero gas for steps for Ozone, or "11,21".

- The forth value/column represents a comma separated list of sequence and step combinations which should be recognized as "precision" gas for this instrument.

- The fifth value/column represents a comma separated list of sequence and step combinations which should be recognized as "span" gas for this instrument.

- The sixth column tells CARBLogger the number of minutes that this instrument reports data in arrears, so it will compensate for that time shift relative to the calibration states of other instruments

- The seventh column represents the manifold or system that an instrument is connected to. If an instrument on system 1 lacks any entry for a sequence and step combination recorded by another instrument on that system, it will be automatically flagged as invalid. Conversely, if an instrument is on a different system/manifold from an instrument which is denoted as using that sequence and step combination, it will not be marked invalid. Any parameter which shares the same system as any other parameter will have its data points flagged invalid when any of the sequence step combinations specified for the other instrument are running. The obvious exception being when that parameter also has the sequence and step flagged as a calibration.

- Note that the presence of RECOV= (line 5) will cause all parameters attached to manifold number 1 (line 5, field 7) to be flagged invalid for sequence/step 15 and 25. Since there is no data labeled RECOV= in the raw text, no data is ever actually reported. Any sequence and step combination which lacks mention in the dms.cfg file will not be reported for the duration of that sequence and step. This is used for Environics 9100 units, as implemented by CARB in order to create a recovery time.

Note: Replacement of an existing instrument with the same instrument (same make and model) does not require any updates to the dms.cfg file. The configuration update is performed only when there is a change in the monitoring equipment setup, such as instrument removal, instrument upgrade, and collocated unit addition, etc.

6.0 REMOTE CONNECTION AND CONTROL

A major benefit of CARBLogger is the ability to control the logger remotely. A remote connection to CARBLogger allows users to control the logger from the office as well as connect to any instrument connected to CARBLogger. To remote into CARBLogger, you will need to have the PuTTY and UltraVNC applications installed on your PC.

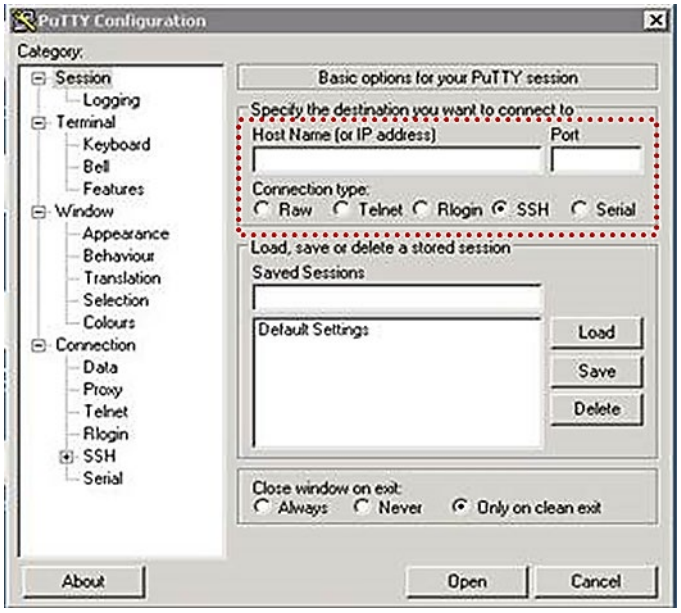
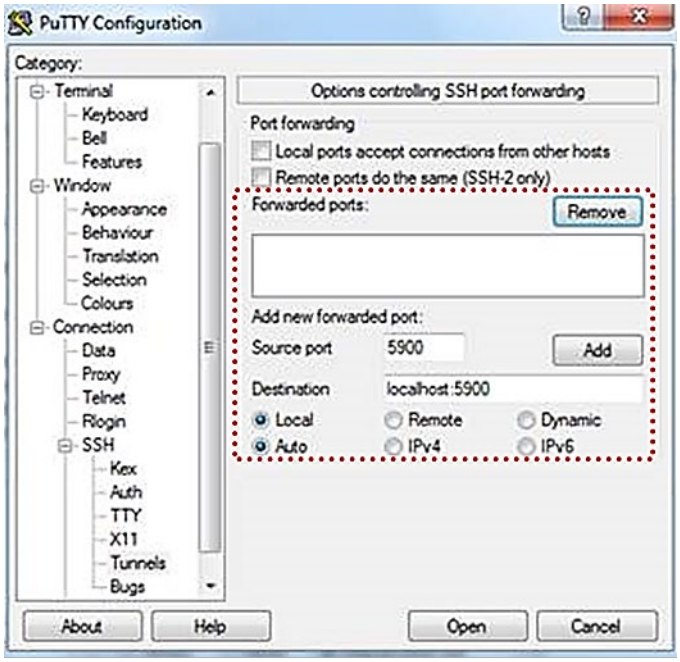
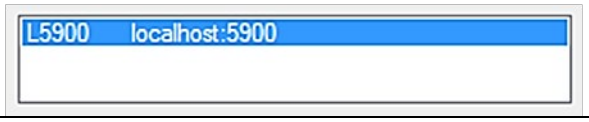
PuTTY is an application that is used to create a secure, encrypted tunnel between the user and the field CARBLogger.

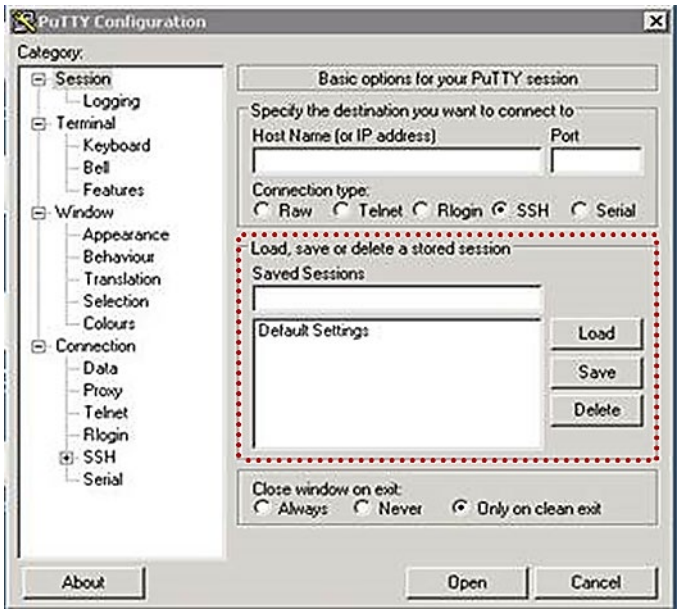
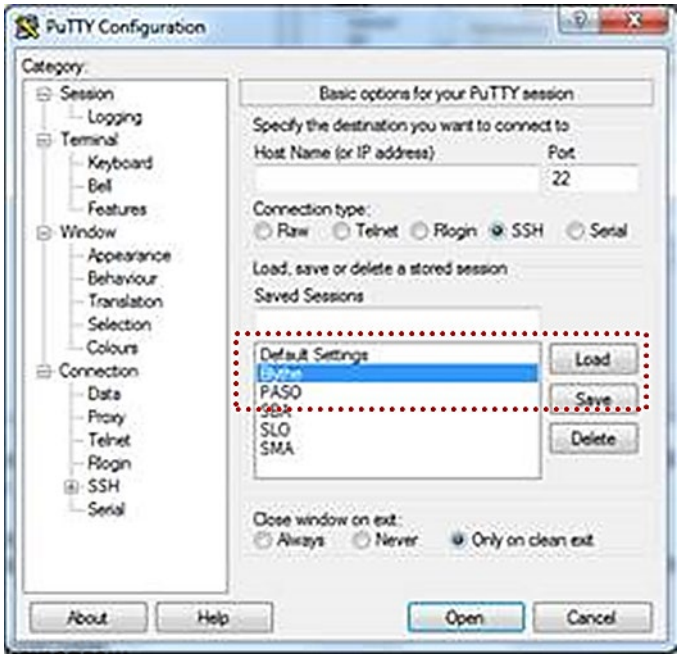
UltraVNC is a powerful, light weight, graphical remote-control application similar to the Microsoft Remote Desktop.

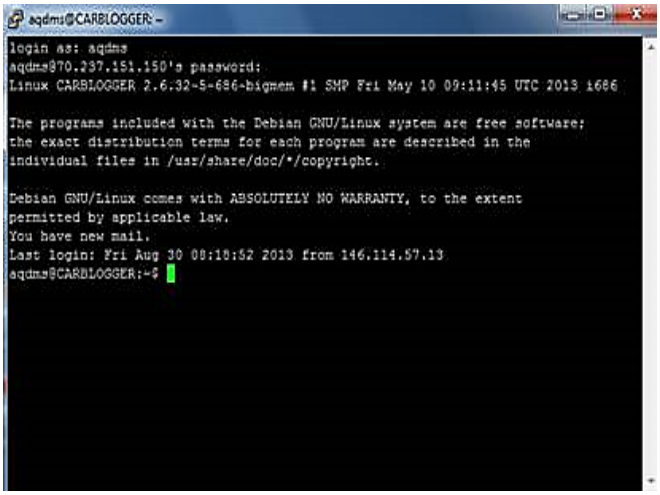
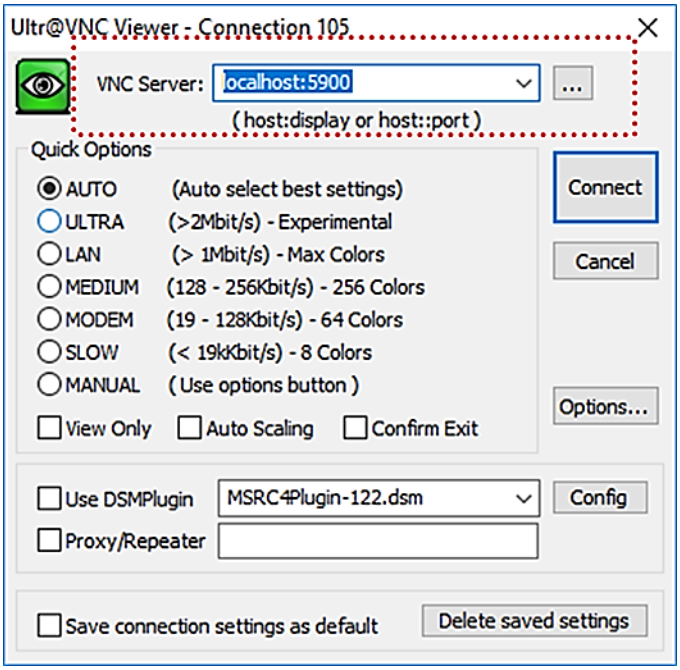
Note: If you do not have PuTTY and UltraVNC on your work computer, please contact OIS to request these applications to be installed onto your PC.

The procedures in Table 6.1 describe how to remotely connect to CARBLogger. If you experience any difficulties, please contact CARBDMS Team for assistance.

Table 7: Remote Connection to CARBLogger (via Putty and UltraVNC).

Step:	Screenshot:
Start PuTTY by clicking the icon on your desktop or selecting the application from the Start menu. Specify the destination by entering a CARBLogger IP Address in the Host Name (or IP address) text box. The default port number will be 22. Change the port number to 242.	
In the Category panel select Connection, SSH, and then Tunnels. (You may need to press the + to expand menu.) This will open the "Options Controlling SSH Port Forwarding" window. Enter this information to the Add New Forwarded Port section and click Add: Source Port: 5900 Destination: localhost:5900 If done correctly, the blank text box labeled Forwarded Ports will then automatically populate a new entry, "L5900 localhost:5900".	 

Step:	Screenshot:
Scroll the Category panel back up and select Session. Type the CL station name (or any name of your preference) into the Save Sessions text box, and click Save. The configuration for your CL Station Connection session will now be saved.	
You may initiate the connection now, or restart PuTTY from your desktop if you have closed this application before. Select the saved profile, click Load, then Open. You will now be connected to your CL destination.	

Step:	Screenshot:
Once connected to the station CL, a terminal shell will appear. Type in aqdms at the "login as:" prompt. Press Enter. At the "password" prompt, enter the station's CL password. For specific site CL password, please contact ODSS. Note: Password entry is masked and invisible to user.	
Start VNC by clicking icon on the desktop or selecting it from the Start menu. In the VNC Server text box enter: localhost:5900 Then press Connect. A remote desktop screen will appear for control of the station's CARBLogger. Note: The VNC server setting will be saved automatically for the next time the VNC is used.	

Once the remote desktop screen has appeared, users may now control the station's CARBLogger as if they are sitting in front of that station's computer.

Note: Users may choose a different connection speed setting for the remote connection session if the site's connection speed does not allow broadband connection. For issues with regard to Internet connection equipment, please contact OIS for assistance.

7.0 BACKFILLING

If the CARBLogger drops, or becomes disconnected from one or more instruments, a gap in the data will appear for the affected instruments. In most cases, this is addressed by obtaining raw data from the instrument's internal data logger (if available), reformatting it for DMS, and sending the formated data to DMS for ingestion.

This process is called Backfilling, and is typically accomplished with the following steps:

1. Establish a digital connection to the instrument via a computer.

2. Query and download the raw text data in its native format for further processing.

3. Reformat the data into a format that will allow DMS to process it.

4. Transmit the data to the DMS ingest server, or use the DMS client to import the data file.

If the instrument is connected via ethernet or serial port to the CARBLogger, all four of the above steps can be accomplished remotely and securely from the remote CARBLogger X-server console.

It must be noted when using raw data imports:

1. Minute data flagging for calibrations or "offline" states will normally NOT occur on these raw text data.

2. The time stamp obtained from the instrument's internal data logging system is less accurate than CARBLogger, unless the instrument has NTP services available.

In the graph below (see Figure 7.1), the red trace represents a correctly time-stamped (NTP synchronized) data stream, while the light orange trace is taken from the raw text data on the same instrument with an internal clock that was 7 minutes slower.

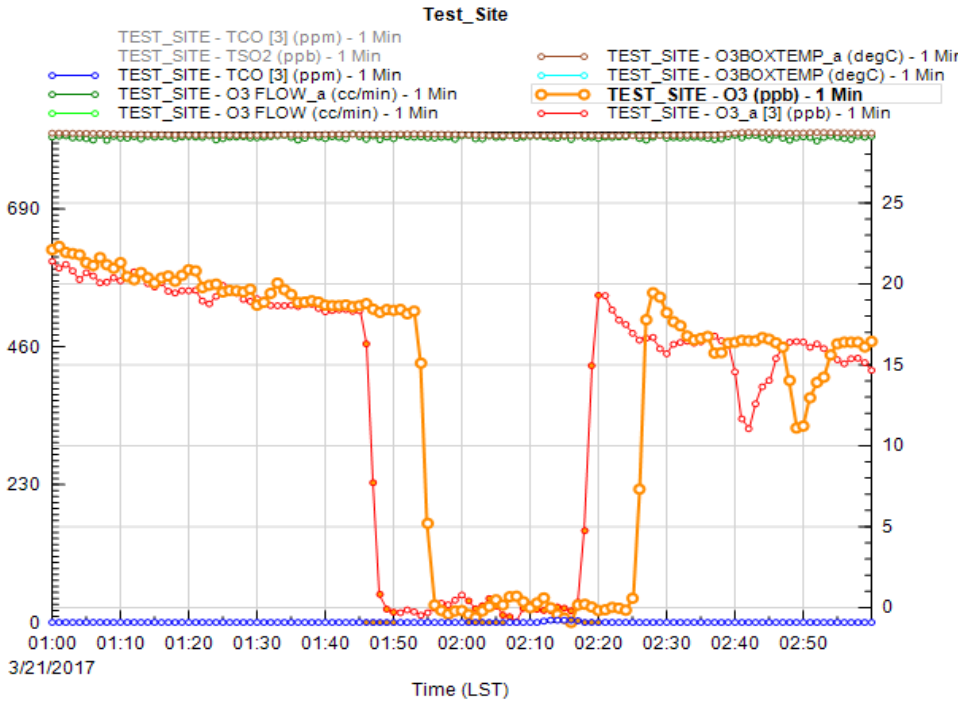


Figure 7.1: Sample Data Time Stamp Issue from Backfilling

Note: It is important to maintain the instrument's onboard clock in the event that data must be backfilled.

In appendices D and E from this SOP, we have provided details on two frequently used examples for data backfilling from instruments. For further understanding on how backfilling is performed, please refer to these appendices.

References:
Appendix D - Backfill Data from the APIT400
Appendix E - Backfill Data from The MET One BAM1020

8.0 ROUTINE SERVICE CHECKS

8.1 General Information:

This section applies to the latest production version of CARBLogger, operating on either the DELL R440, R430, R620, or T610 server chassis. Since these form factors are industrial grade servers, regular maintenance needs to be applied to the hardware, the file system, operating system, applications, and CARBLogger code base.

8.2 Daily Checks:

Each morning, confirm that the DMS status screen for your station appears green and that you have received your errors and alert summary emails.

Optionally, leave CARBLogger in the data display screen, and confirm that all collected parameters appear black.

For instructions on how to check the DMS status for your station, please refer to the DMS SOP document.

8.3 Weekly Checks:

Confirm that the NTP processes are properly functioning by comparing the clock on the CARBLogger desktop to your cell phone or some other time standard. If they do not match, please contact ODSS for assistance.

If the "bezel" display on the front of the server appears orange or amber, (alternatively, confirm that the background of the display is not blue), please make a note of any error messages, and notify CARBDMS TEAM immediately.

Check the CARBLogger data display screen and confirm all parameters appear black.

8.4 Biweekly Checks:

The servers on which CARBLogger run have hot-pluggable, RAID 1 disk configurations. As a result, it is possible that one disk may fail, while data clients remain unaware. It is critical that such a condition is detected early so that the compromised disk can be rebuilt.

Remove the front face of the CARBLogger once per month and confirm that the lights on each of the two disk drives are intermittently green. If one of the disks

has amber status lights, contact CARBDMS TEAM immediately to obtain a replacement hard drive or have the current one rebuilt.

8.5 Monthly Checks:

Pay attention to the data drive disk space value from the nightly email summary you receive. If the data drive gets larger than 200 GB, the data drive may need to be cleaned up. Notify CARBDMS TEAM for this maintenance task.

The servers on which CARBLogger run have redundant power supplies. If one fails, a text message may appear on the LCD display indicating the loss of a power supply. Alternatively, look at the LEDs on each power source on the back of the chassis to confirm that both lights on both power supplies are steady green. If you find that either of the power supplies has failed, contact CARBDMS TEAM immediately for a resolution.

8.6 Annual Checks:

Remove the cover of the CARBLogger chassis and blow out the inside with clean, dry, compressed air.

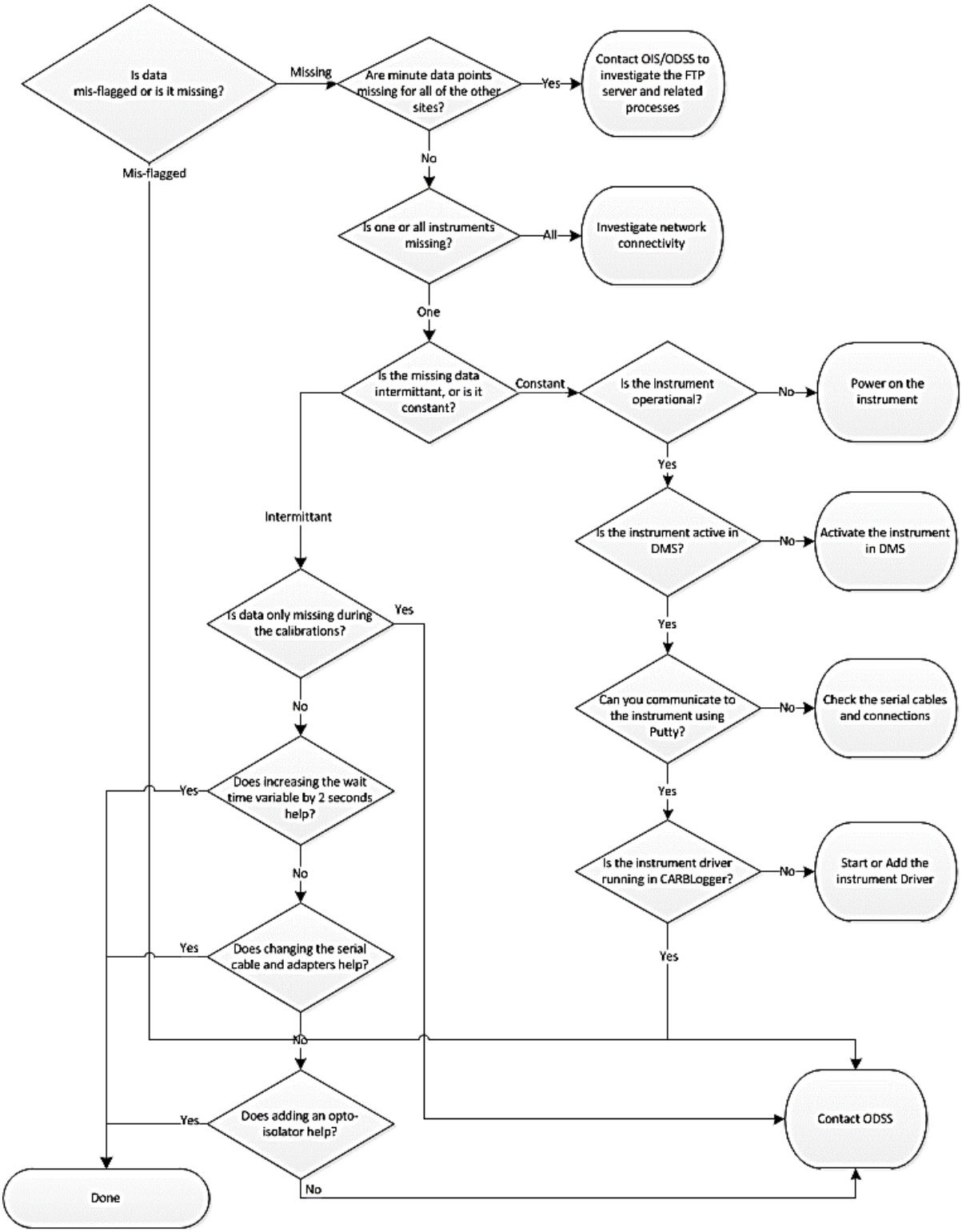
For purposes of data continuity and system security, most CARBLogger system updates and manufacturers released firmware updates are not applied until they have been fully vetted on the test site by CARBDMS TEAM. These updates would normally be applied annually and by CARBDMS TEAM or OIS staff only.

9.0 GLOSSARY/LIST OF TERMS

Term	Description
Back poll	Process the raw text data to generate DMS data files, or obtain raw text data from the instruments and convert that into DMS formatted data.
Calibration Flagging	With the exception of sites running the API 400 IZS models, AQSB air monitoring stations use auto-calibrators which control the station calibrations. The sequence and step number of each calibration routine determines how data get flagged for DMS. If you remove or change a sequence, refer to the "Update the DMS.CFG file" procedure to reflect your station changes in the dmsout process. If you forget to do this or do it after the fact, you may use DMS to change your calibration flagging at a later time.
Driver	CARBLogger is actually a collection of mini data-loggers (or drivers), with each driver designed to acquire data from a particular instrument. The CL interface allows runtime parameters for each driver to be edited and for the drivers to be started, or stopped.
DMS	Data Management System (DMS) is used to analyze and store ambient data. A reporting mechanism for DMS which converts the raw text stored by CL into flagged data for ingest by DMS was written. To use this process users would need to update the dms.cfg file.
Dms.cfg	Dms.cfg is a file used by the Dmsout process to calculate automatic data flagging for DMS.
Dmsout	Dmsout.sh is the reporting process which was developed to produce flagged one-minute data for DMS.
Ingest	CARBLogger natively stores all data as raw ASCII text. Ingest is the term used to describe the process of pulling this raw text into a database. The advantage of ingesting data in to a database is that much more complex data analysis procedures can be run against the data.
Interface	This is the portion of CARBLogger that allows users to easily configure and control drivers. For most users, this is the only portion of CARBLogger they will need to be familiar with. Refer to section 3.2.
Instrument	The term "Instrument" is used interchangeably with the term

Term	Description
and/or Channel	"Channel". All of the data collected from a single instrument is collectively referred to as a channel. In the "Display Channels" screen, a single channel will share a common instrument name. Refer to section 3.8. For example, the NOX channels all came from the "API200E" driver.
NTP	Network Time Protocol (NTP) is used by CARBLogger to keep its clock synchronized with NIST time servers. In this way, the time stamp affixed to raw data is kept accurate and precise. Typically, we use NIST time servers and are set to GMT -8.
Raw Data	CARBLogger acquires raw text from all instruments and, for the most part, records it "as is" in ASCII format. The only modification it makes is to prepend the time and date stamp obtained from the NTP server. The data from every parameter and instrument is stored in plain text. This is referred to as "Raw Data".
Root Terminal	A root terminal, much like a normal terminal, is simply the Linux analog of the "DOS prompt", except that it is being run as the local administrator. CARBDMS TEAM may ask you to perform some operations using this method. To access the root terminal, you will click on Applications/Accessories/Root Terminal. You will be prompted for the root password, for this CARBLogger.
Spmout	Spmout.sh is one process which runs a report against the on board MySQL database to produce a text based, CSV formatted pivot table of all data from any CARBLogger for the previous day. Most station operators receive a rollup of the previous days' worth of data in this format each morning via email. Unlike with Dmsout, any null values will yield a zero.
Terminal	CARBLogger is running on a Linux operating system. The terminal (or shell) is analogous to the "DOS prompt" in Windows. On the server versions of CARBLogger, you can open a terminal by double clicking on the "CARBLogger" icon located on the desktop.

Appendix A: TROUBLESHOOTING DIAGRAM



Appendix B: CARBLOGGER MAINTENANCE CHECK SHEET

AQSB MONTHLY QUALITY CONTROL
MONTHLY MAINTENANCE CHECK SHEET 605
CARBLogger

Location:_____ Month/Year:_____
Station Number: Technician:
Property Number:_____ Agency:_____

Dates				
NTP Check				
All Values Black				
Tuesday Morning Gap Present				

Operator Instructions:

1. Daily: Review data status in DMS.

2. Weekly: Confirm a five minute gap each Tuesday morning, at 0130
Confirm all data "black" and on time.
Confirm that the clock on the CARBLogger matches a cell phone. If it does not, contact CARBDMS TEAM.

3. Bi-weekly: Date: _____ Drive one color: ____ Drive two color: ____
Date: _____ Drive one color: ____ Drive two color: ____

4. Monthly: Data Drive Space: _____
Power Supplies: _____

5. Yearly: CARBDMS TEAM maintenance performed _____

Date	Comments/Maintenance

Reviewed by: _____ Date: _____

AQSB QC Form 605 (CARBLogger) (10/17) Page 1 of 1

Appendix C: INSTRUMENT COMMUNICATION CONFIGURATION

After the site settings have been confirmed, take the following steps to ensure that each connected instrument has been properly configured.

- Connect the instrument to an empty serial port on the CARBLogger squid (8-Port Serial Cable).

- If the serial card is a Perle variant, the serial port your instrument is connected to will be assigned /dev/ttyPS[x], where x is one less than the number tag on the cable. On the virtual layer, port number starts off from zero, or ttyPS0.

- If the serial card is a StarTech variant, ensure that CARB applied labels are present. Due to the manufacturing process, /dev/ttyS[x], is used instead.

  For example, if instrument is connected to cable number 6, this means that the computer port we need to speak to is /dev/ttyPS5.

- Follow the instrument operations manual specific to your instrument in order to activate the serial port and set the baud rate.

- From the Gnome/CARBLogger desktop choose **Applications/Internet/Putty SSH Client**. Refer to Figure C1 below.

- Tick the **Serial** radio button and enter in the serial port and baud rate of the device you wish to communicate with, then press on the [Open] key. Refer to Figure C2 below.

- After you have successfully managed to communicate with the instrument of interest, make a note of the communication settings being used. These are the settings to be entered into CARBLogger interface as detailed in the Add Instrument configuration section (Sections 3.4 and 4.3).

WARNING: Be sure to close this putty session before attempting to have CARBLogger communicate with the same serial port. Failure to close any serial program running on any serial port will cause the CARBLogger to experience a gap in data communications for as long as the program remains open

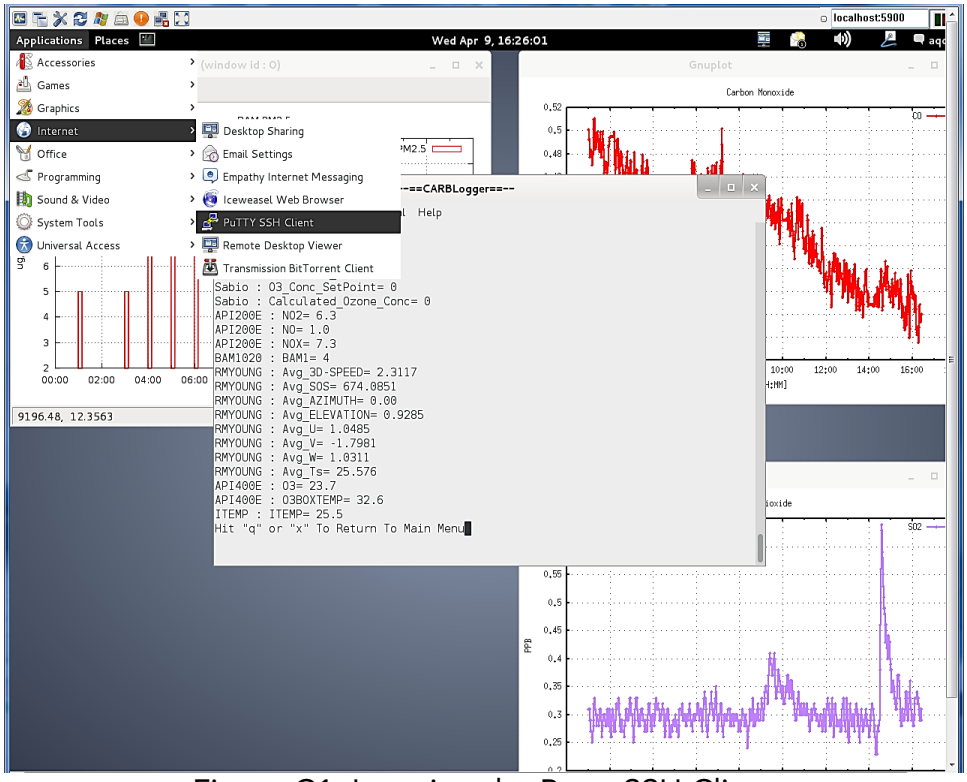


Figure C1: Locating the Putty SSH Client

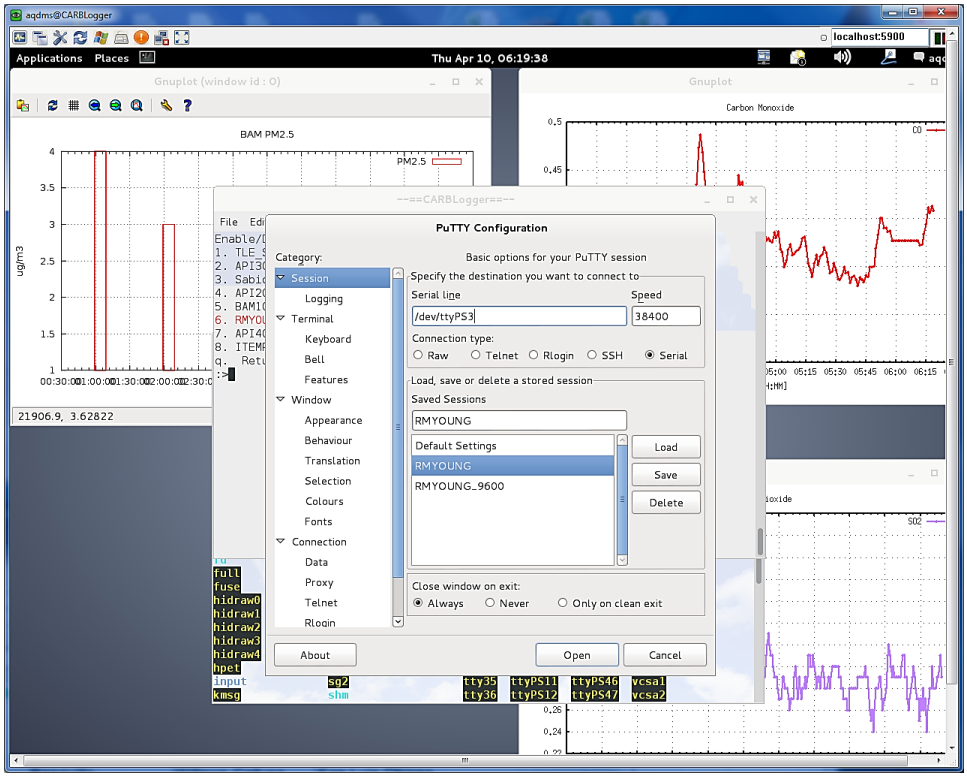


Figure C2: Setting up the Putty SSH Client

Additional information for Putty SSH Client Configuration:

- **Ready to Send/Clear to Send (RTS/CTS) flow control** should be set to OFF on all instruments. A limitation of the Perle Ultra Serial card driver (as installed in the CARBLogger chassis) is that all serial connections either must uniformly use RTS/CTS. It is not possible to connect two instruments to the Perle serial card if one requires RTS/CTS and the other does not, without experiencing connectivity issues. If an instrument must have RTS/CTS enabled, please use the chassis serial port in the back of the unit (/dev/ttyS0). You may choose this serial connection from CARBLogger by choosing "/dev/ttyS0".
  Finally, if you have more than one instrument requiring this flow control, it will be necessary to have CARBDMS TEAM recompile the Perle driver, and you will have to confirm that all instruments being monitored have this serial attribute enabled for each Perle serial card used.

- **Baud Rate:** The most tested baud rate in our network is currently 9600, although CARBLogger and instruments can use various baud rates. As a general rule, the longer the serial cable and the higher the baud rate, the more errors that data transmissions are likely to contain. Because we have successfully used this baud rate for various instruments at lengths exceeding 300 feet (well beyond the published specifications for RS232), we recommend using 9600 unless there is a requirement for doing otherwise.

- **Firmware Versions:** Most CARBLogger drivers have been written using the currently available firmware, as the instrument was configured at the time that the driver was written. If you are applying an update to an instrument's firmware, or changing the configuration of an instrument which was not used at the time that the CARBLogger driver was written, monitor your raw data output closely. Changes in firmware or internal data logger configuration can cause changes in the unit's slope, offset, and error handling procedures used to report data.

Appendix D: BACKFILL DATA APIT400

If a data logging system becomes disconnected from the instruments in a station, it becomes necessary to acquire raw data from the instrument's internal data logger, and convert it into a format that DMS could ingest. In the below site, data acquisition stopped between 20:00 and 08:00. While the techniques needed to modify raw data are not discussed here, we demonstrate backfilling data from the APIT400.

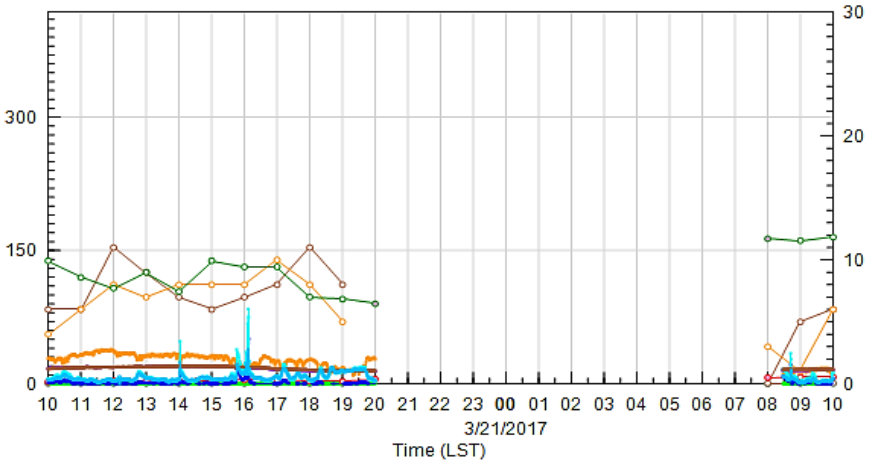
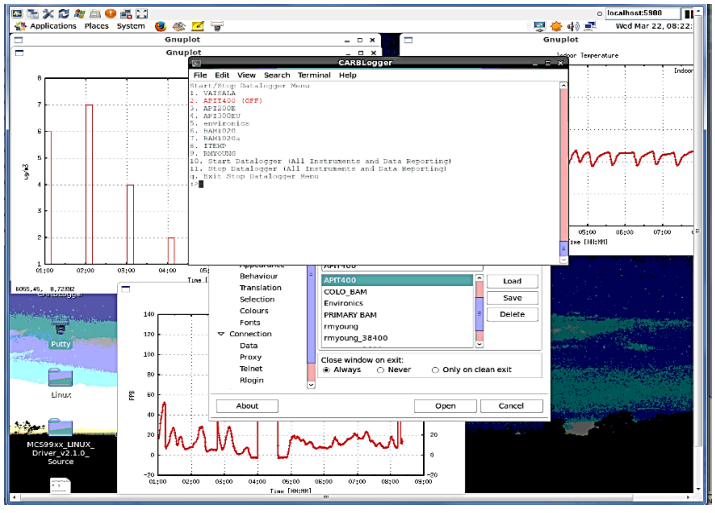
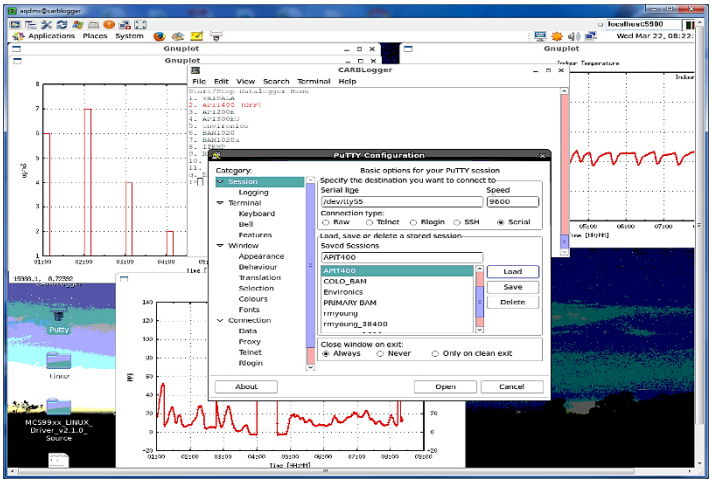
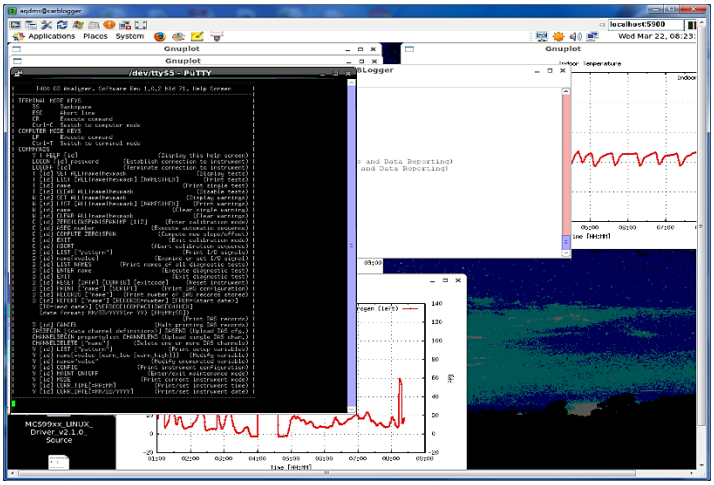
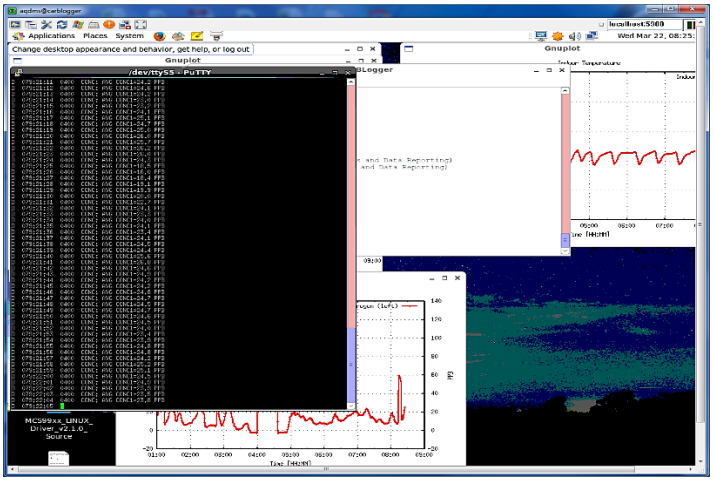
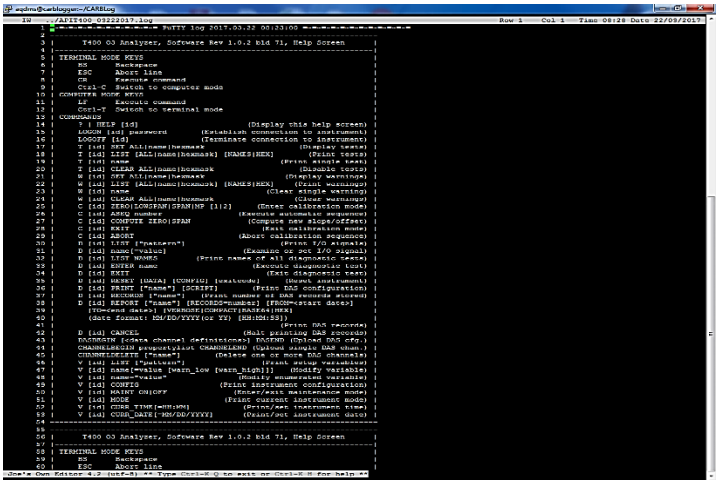
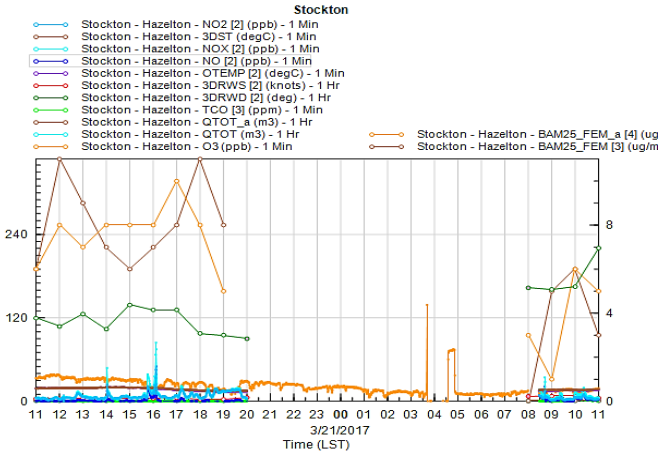
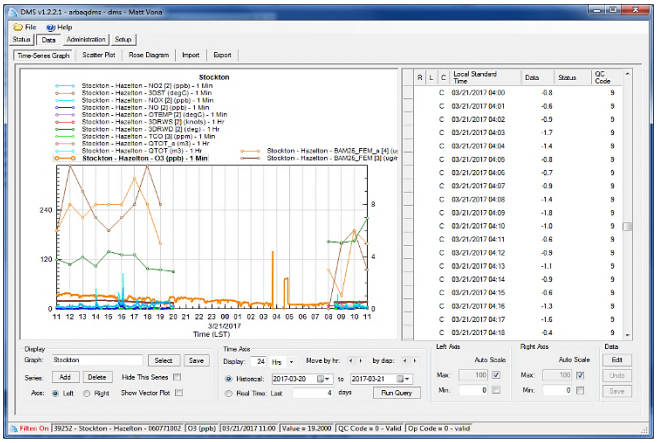


Figure D1: Data gap between 20:00 and 08:00 hours

Step:	**Screenshot:**
Since the instrument and driver are using a serial port, it is necessary to first stop the driver from the Start/Stop Instruments/Datalogger menu.	

Step:	Screenshot:
Open Putty or some other serial capable program, and connect to the instrument. Be sure to configure the program to record your session.	
Query the instrument's internal data logger.	
After the data has been downloaded, close the serial client and restart the driver.	

Step:	Screenshot:
Edit the downloaded data (in this case) by removing the headers and any non-data containing lines.	
Run the script against the data, and check DMS in 15 minutes. *Refer to Section 7.0 for what script to use.	
Mark any calibration periods as invalid, or as calibrations, so as to preserve real time data exports.	

Appendix E: BACKFILL DATA MET ONE BAM1020

When back filling data from any instrument, it is necessary to be familiar with both the instrument and the in-situ configuration of its data logger. The BAM25 has customizable data labels, the ability to record data in either micrograms per cubic meter or milligrams per cubic meter (on some models), and which columns to capture, among other localizable variables. The slightest change to this configuration can have dramatic consequences to accuracy and functionality of scripts, spreadsheets, or other mechanisms used to prepare data for DMS ingest.

Below is an example of data taken from the internal data logger of a Met One BAM25 as configured in Site A:

Time,Conc(mg/m3),Qtot(m3),WS(KPH),WD(DEG),IT(C),RH(%),Delta(C),AT(C),E,U,M,I,L,
R,N,F,P,D,C,T,
03/20/17 00:00,0.007,0.700,61.1,4.4,4.4,26,0.468,14.0,0,0,0,0,0,0,0,0,0,0,0,0,

Below is an example of data taken from the internal data logger of a Met One BAM25 as configured in Site B:

Time,Conc(mg/m3),Qtot(m3),WS(KTS),WD(DEG),IT(C),RH(%),Delta(C),AT(C),E,U,M,I,L,
R,N,F,P,D,C,T,
03/20/17 00:00,0.007,0.700,61.1,4.4,4.4,26,0.468,14.0,0,0,0,0,0,0,0,0,0,0,0,0,

Any automated process which expects the fourth column title to be (KPH) instead of (KTS) may fail, depending on how it is written, and vice versa.

At the time of this writing, DMS records and reports PM2.5 data in micrograms per cubic meter, while all BAM samplers in the network presently store this data in milligrams per cubic meter. As such, backfilling this data will require multiplying the "Concentration" value by 1000 prior to submitting to DMS.

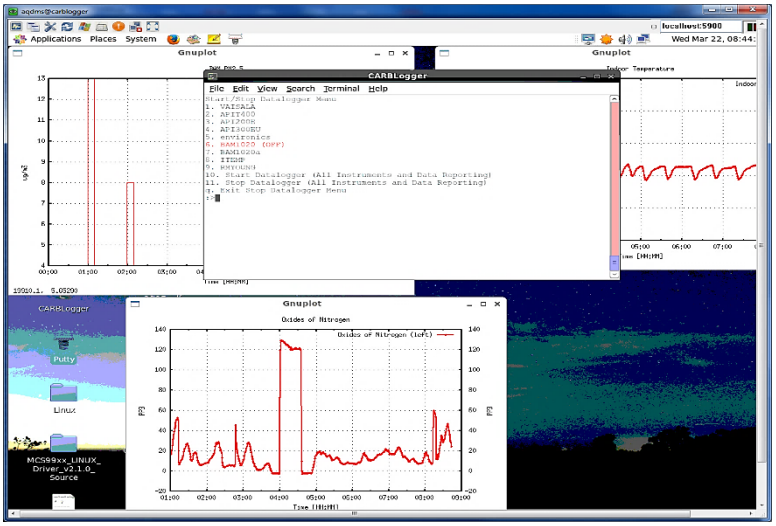
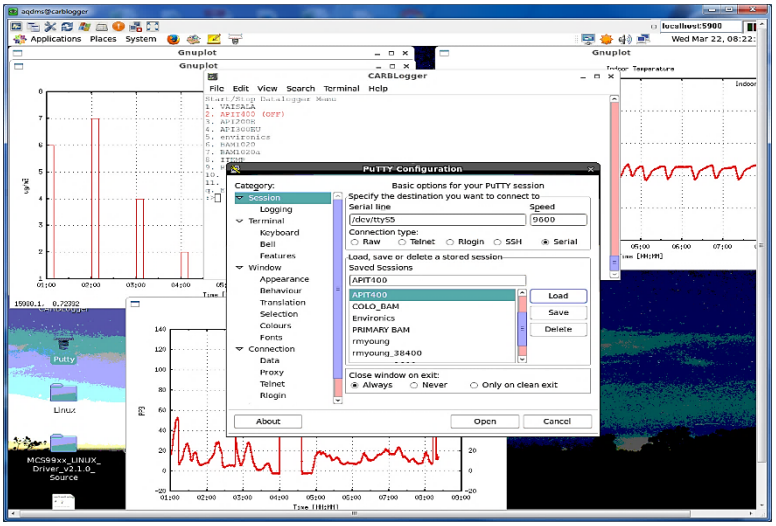
If any of the last error columns contain a one, it may indicate an error condition for that hour. For this reason, it is advisable to invalidate or qualify (mark an Op code of 5) any data collected for any such hours.

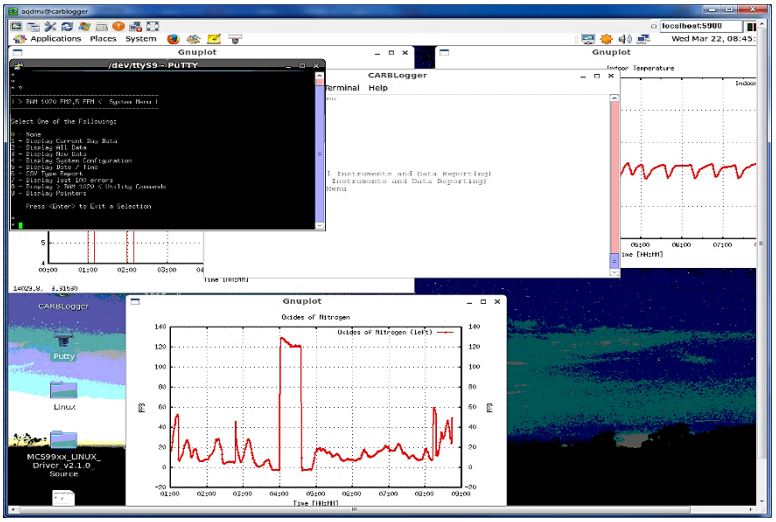
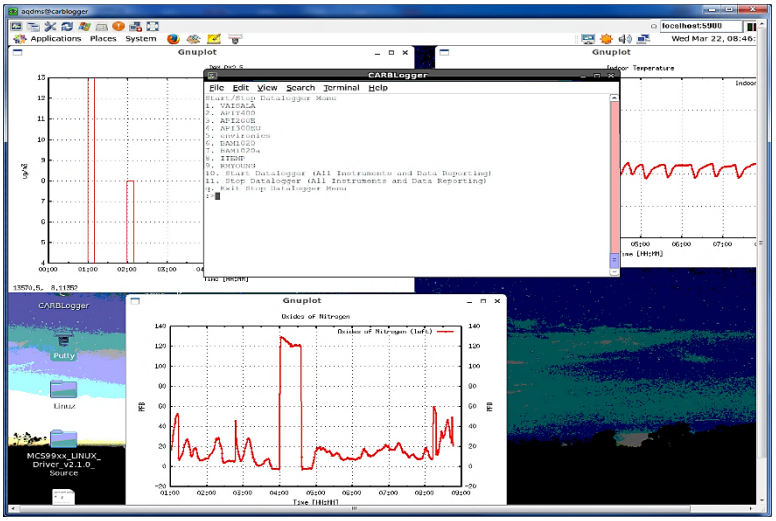
The internal date stamp in the raw data is applied by the Met One BAM at the time that the previous hour value is calculated, not for the beginning of the hour that the data actually represents. As a result, any automated mechanism for importing this data needs to subtract an hour from the time stamp. So, in the above example, the correct date stamp *3/20/2017 00:00* should become *3/19/2017 23:00*.

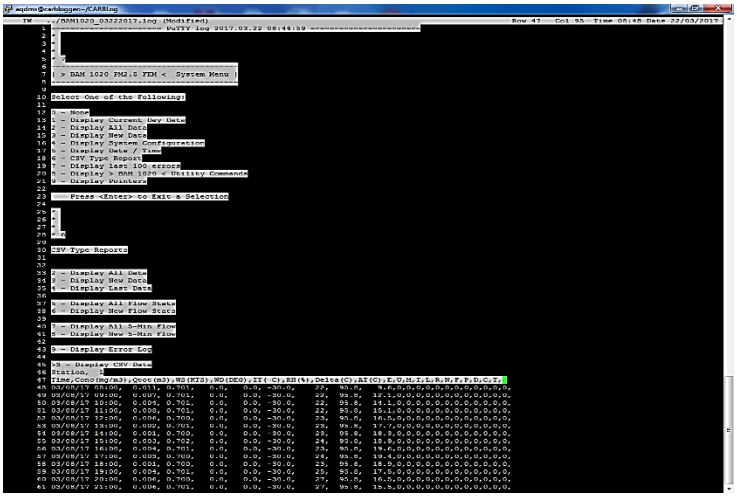
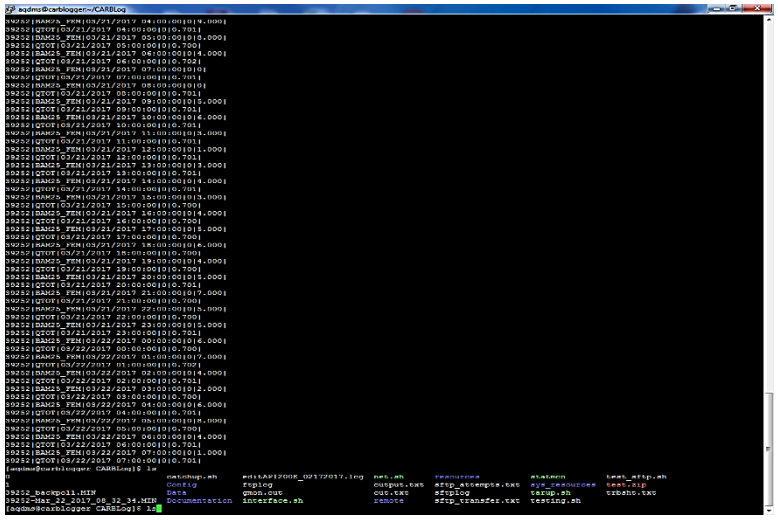
It should also be noted that, over time, these subtleties will be changed by the

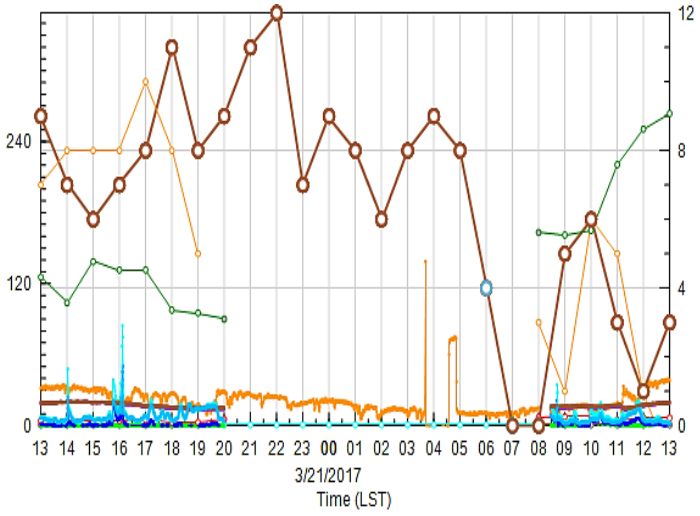
manufacturer. For instance, on newer BAM samplers, it is now possible to have them store the data in microgram per cubic meter, and to have them date stamp correctly. If these changes are applied, extra care must be taken into account in order to change/update all BAM instruments within the network.

In this example, we use a script written by CARBDMS TEAM made available on most CARBLogger to import raw data from the BAM1020 PM2.5 monitor into DMS.

Step:	Screenshot:
Stop the BAM driver from the Start/Stop Instruments/Datalogger menu on the user interface.	
Open Putty or some other serial capable program, and connect to the instrument. Be sure to configure the program to record your session.	

Step:	Screenshot:
Query the instrument's internal data logger. Download data by pressing 6, 4 to download all new data in a comma separated format.	
After the data has been downloaded, close the Putty serial client, and restart the BAM driver.	

Step:	Screenshot:
Edit the downloaded data file by removing all non-data lines. In this case, remove all headers.	
Having downloaded the data from step 4, use the "sys_resources/scripts/bam25_2dms_stockton.sh" script to process the data extract.	
This script in particular performs labeling to the DMS data (i.e., BAM25_FEM, or BAM25_FEMa, or BAM25), and adjusts for site specific data logger alterations.	

Step:	Screenshot:
Check DMS in 15 minutes, and confirm that the data has been backfilled.	

Appendix F: LIST OF SOFTWARE DEPENDENCIES

Software	Additional Information
Bourne Again Shell (BASH)	http://www.gnu.org/software/bash/
CARBLogger	http://code.google.com/p/carblogger/
Debian	http://www.debian.org
Feed Gnuplot	https://github.com/dkogan/feedgnuplot
GAWK	https://www.gnu.org/software/gawk/
Gnome	http://www.gnome.org
Gnuplot	http://www.gnuplot.info
GoogleCode	http://code.google.com/
Linux	http://www.linux.org
Maria DB	https://mariadb.org/
Mutt	http://www.mutt.org/
Open SSH	http://www.openssh.com/
Sed	https://www.gnu.org/software/sed/manual/sed.html
Sjinn	http://sjinn.sourceforge.net/
WVDial	https://code.google.com/p/wvdial/
X11VNC	http://www.karlrunge.com/x11vnc/

Appendix G: CARBLOGGER AND CARB SITE NUMBERS

DMS Site Number	Station Name
04633	Paradise – Theater
03614	Jackson - Clinton Road
04625	Chico – East
04636	Gridley - Cowee Avenue
04638	Paradise – Airport
05633	San Andreas - Gold Strike Road
06646	Colusa - Sunrise Blvd
09690	Placerville - Gold Nugget Way
09691	South Lake Tahoe - Sandy Way
10251	Fresno – Garland
11676	Willows – Colusa
13698	Calexico - Ethel Street
15242	Edison
15243	Oildale - 3315 Manor Street
15248	Shafter - Walker Street
15249	Arvin - Di Giorgio School
15252	Mojave - 923 Poole Street
15255	Bakersfield - California Avenue
22742	Yosemite Village - Visitor Center
31822	Roseville - N Sunrise Blvd
33201	Blythe - Murphy Street
34305	Sacramento - T Street
39252	Stockton – Hazelton
40836	San Luis Obispo - Higuera Street
40850	Paso Robles - Santa Fe Avenue
42408	Santa Maria - South Broadway
50568	Modesto - 14th Street
51898	Yuba City – Almond
51899	Sutter Buttes S. Butte (seasonal)
52910	Tuscan Butte (seasonal)
54568	Visalia - N. Church
55930	Sonora - Barretta Street
57577	Davis - UCD Campus

Appendix H: DESCRIPTION OF CARBLOGGER GRAPHS

All graphing capability is built into individual drivers as another data product. Most CARBLogger drivers use an open-source command-driven function and data graphing system called Gnuplot (except the RM Young 81000 which uses python).

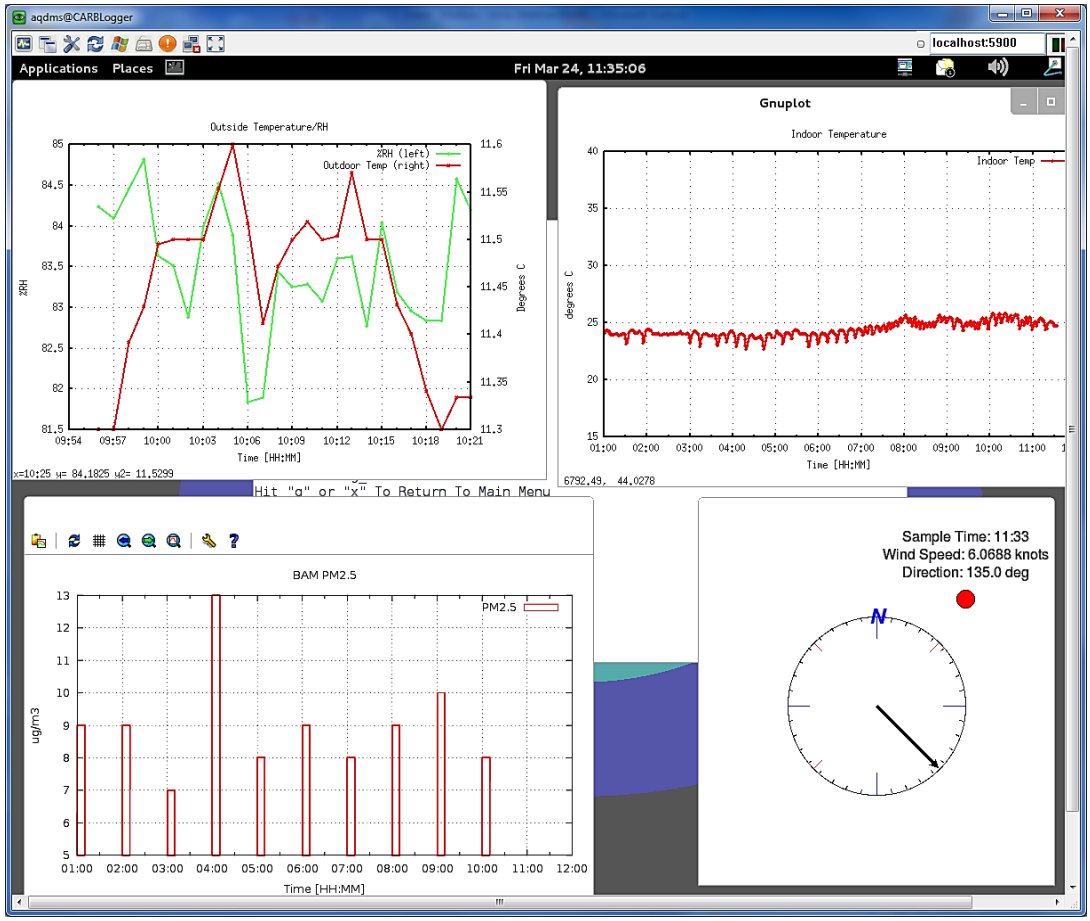


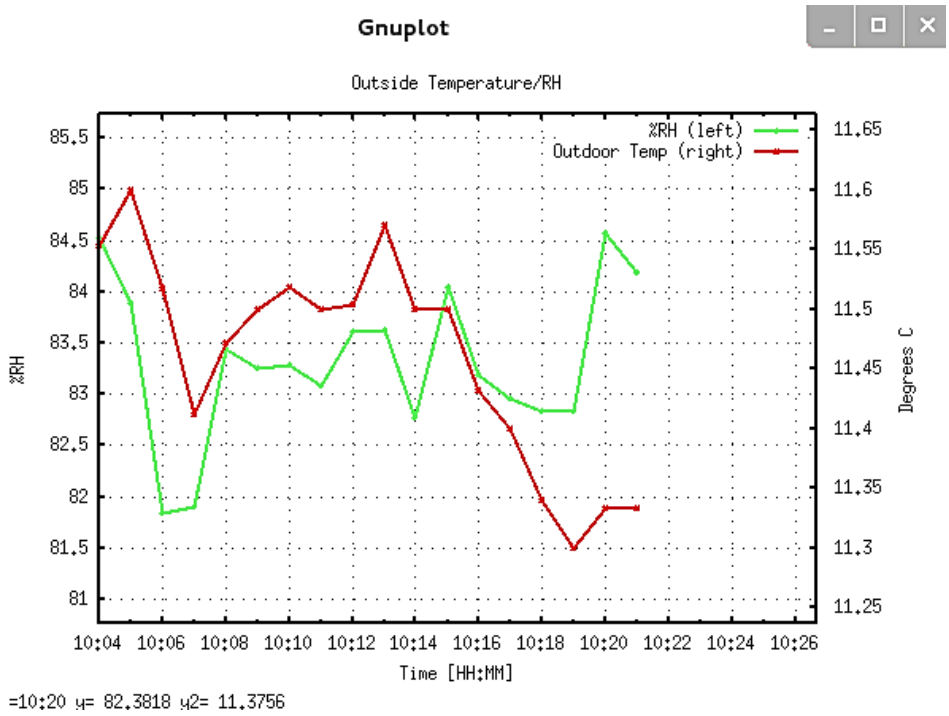
Figure H1: Different Data Visual Displays on CARBLogger

By convention, graphs for each instrument start and stop with each driver and plots minute data from 0100 each morning to the current time on the 25th second of each minute. The Gnuplot graphs are plotted each minute using the raw data that each driver collects, every minute. If you note that a graph has stopped updating, the easiest way to get it to refresh is to simply stop and restart the driver using the text interface.

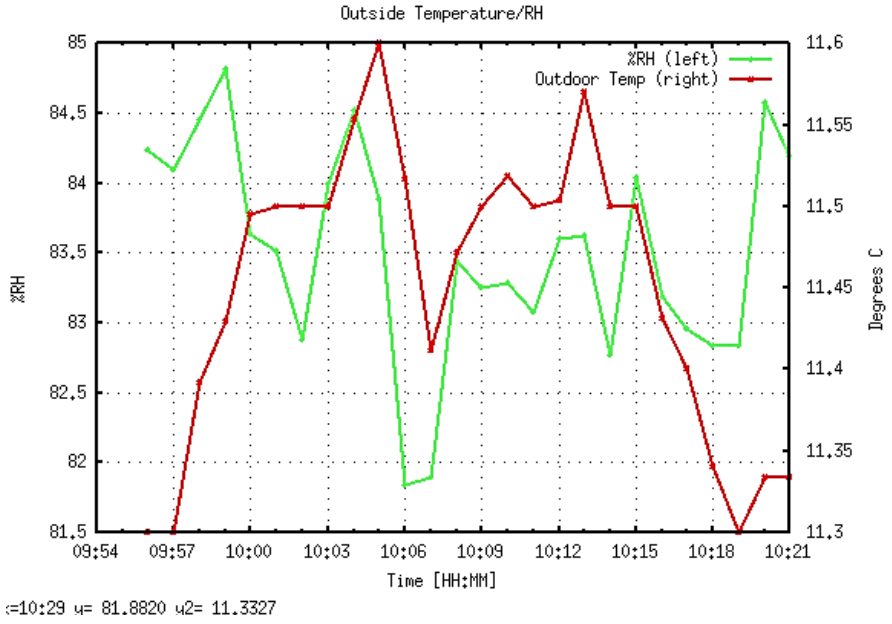
Graphs on CentOS are non-interactive. When graphs are running on the Debian operating system, however, these graphs all have the following additional capabilities:

1. Holding the [Shift] key while pushing the middle mouse forward or backward

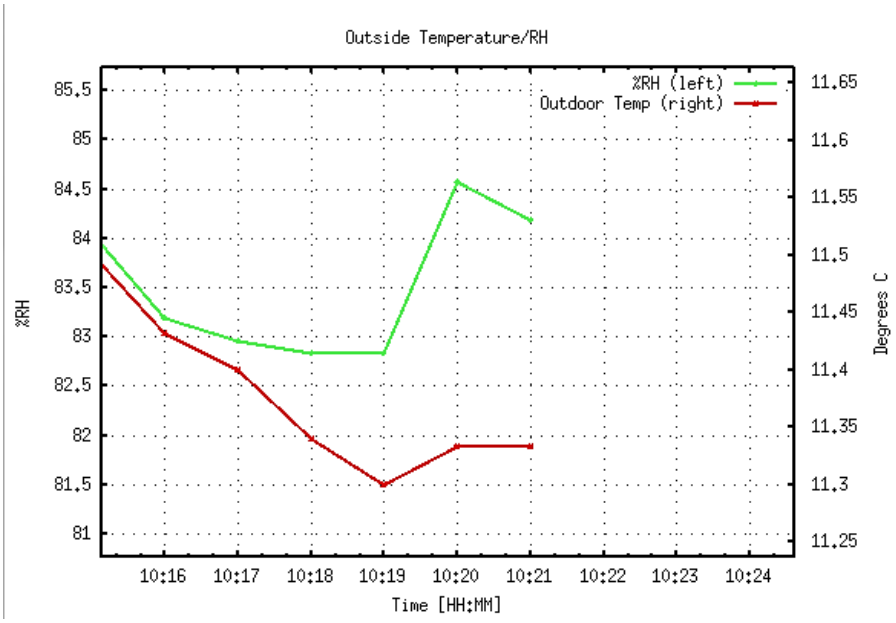
will cause the graph to translate to right and left, respectively.



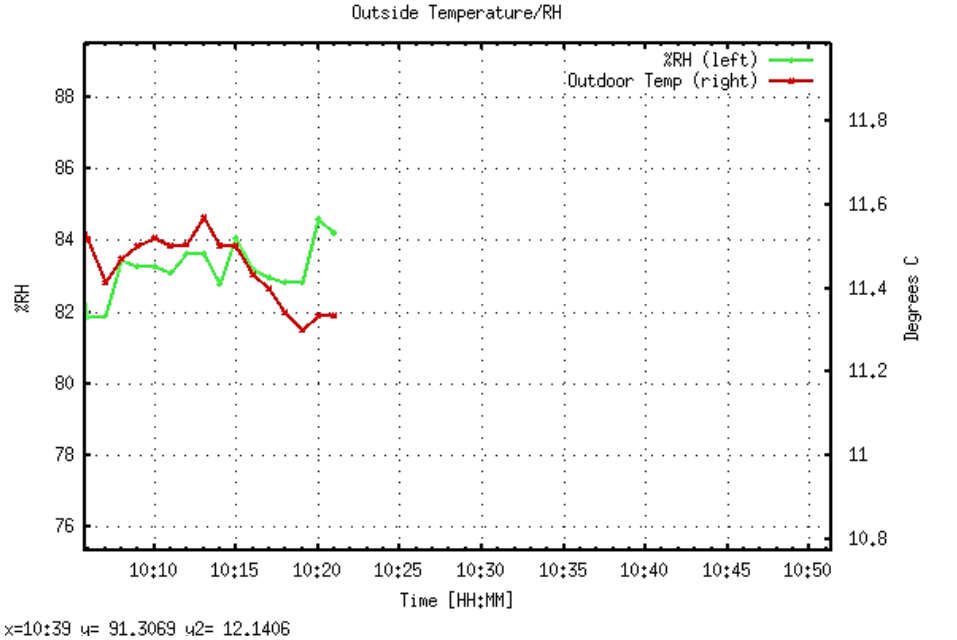
2. Holding the [Alt] key while pushing the middle mouse forward or backward will cause the graph to translate up or down, respectively.



3. Holding the [Ctrl] + [Shift] key while pushing the middle mouse forward or backward will cause the graph to expand or compress along the X axis, respectively.



4. Holding the [Ctrl] + [Alt] key while pushing the middle mouse forward or backward will cause the graph to zoom in or out, respectively.



Appendix I: OPERATIONAL CODES ASSIGNED BY CARBLOGGER

OP Code	Definition	Description
0	Valid	This is a valid data point, and it will be aggregated into larger valid time period averages by DMS.
*1 (11, 21, 31)	Zero	This is a valid data point, and it will be aggregated into larger, valid contiguous zero averages occurring in the same time periods by DMS.
*2 (12, 22, 32)	Gas Precision	This is a valid data point, and it will be aggregated into larger, valid contiguous precision averages occurring in the same time period by DMS.
*4 (14, 24, 34)	Gas Span	This is a valid data point, and it will be aggregated into larger, valid contiguous span averages occurring in the same time period by DMS.
**9	Invalid	This is an invalid data point, and it will map with a QC code (9) that excludes the data from aggregating into valid time averages in DMS.
**51	Maintenance	This is an invalid data point, and it will map with a QC code (22) that excludes the data from aggregating into valid time averages in DMS.
**52	Instrument Repair	This is an invalid data point, and it will map with a QC code (22) that excludes the data from aggregating into valid time averages in DMS.
**53	Off-Line	This is an invalid data point, and it will map with a QC code (20) that excludes the data from aggregating into valid time averages in DMS.

OP Code	Definition	Description
**60	QA Audit	This is an invalid data point, and it will map with a QC code (14) that excludes the data from aggregating into valid time averages in DMS.
**64	BAM Zero Test	This is an invalid data point, and it will map with a QC code (13) that excludes the data from aggregating into valid time averages in DMS.
**65	Autocal Off Phase	This is an invalid data point, and it will map with a QC code (10) that excludes the data from aggregating into valid time averages in DMS.
**68	Calibration	This is an invalid data point, and it will map with a QC code (11) that excludes the data from aggregating into valid time averages in DMS.
**69	Auto Calibration	This is an invalid data point, and it will map with a QC code (10) that excludes the data from aggregating into valid time averages in DMS.

* Depending on the time of day that the calibration event is accomplished, a numeric value may be inserted prior to the Op code. No preceding value indicates that the calibration took place between the hours of (0:00- 5:59), and values of 1, 2, and 3 indicate that the calibrations took place in the time periods of (6:00-11:59), (12:00-17:59), and (18:00-23:59) respectively. So, an Op Code of 34 is a span which took place between after 1800 hours and before midnight.

** If more than 25% of the data within the hour are invalid, a null code corresponding to the mapped QC code will be assigned to the hourly data point.

Appendix J: HISTORY OF CARBLOGGER

Experimental –

The first version of CARBLogger was developed on Ubuntu to assist the Special Purpose Monitoring section to obtain telemetry for two co-located RM Young 81000 Ultrasonic anemometers running on top of the CalEPA building on 10th and I street. The implementation allowed remote configuration of the instruments, and recorded all data in a time-stamped format. No user interface was available with this version of CL.

Diagnostic Data System –

Using the experimental scripts, after having run consistently for over a year in adverse conditions, CARBDMS TEAM has developed a digital diagnostic data system, using old MLD work stations and both Linux Mint and Debian operating systems. The user interface, several drivers, and the ability to parse data streams and email errors were added. An installation script was also developed to convert any Ubuntu distribution into a CARBLogger.

Experimental Data Logger –

After having run successfully for several years as a diagnostic system on old equipment at several sites, the Special Purpose Monitoring section requested additional function from the logger; a one-minute pivot table containing all data collected by the logger. A MySQL database and procedures to parse, ingest, and report this data was added.

CARBLogger v1.0 *(Production Data Logger)* –

The first version of the CARBLogger used for production data logging was developed for the implementation of the NCore air monitoring station in Fresno, California. U.S.EPA NCore requirements included digital data acquisition due in part to the implementation of trace air monitoring instruments and need for MDL determination. The site had two manifolds, and a direct outlet for one trace instrument. Additionally, any digital system needed to be able to run in parallel with the existing analog system, so available status contacts and analog outs were not always available. At this point in time, no other full digital data acquisition system was available for us to use with DMS and with our instruments.

This version of CARBLogger added stored procedures to the MySQL database (from the experimental data logger) which were capable of allowing an operator to describe how a site is configured. These stored procedures (based on the operator configuration) then produce correctly flagged and formatted one-minute data for DMS ingest, and ftp them to the CARB ftp servers for pickup by DMS. Also, back poll

functionality was created so that CARB staff could request a resubmittal of data.

CARBLogger v1.2 –

The second version of the CARBLogger system was based on the Debian OS. In addition, the internal database in the logger was changed from MySQL to MariaDB. The following changes were made:

1. The user interface and all drivers were updated allowing the user to input other non-operational states including:
    a. Enable
    b. Off-line
    c. Routine-Maintenance
    d. Instrument Repair
    e. Manual Calibration
    f. QA Audit
    g. Zero Test
2. The number of infrequently used instrument drivers in the production code repository/interface was decreased.
3. Most drivers were updated to interactively plot real time data using Gnuplot.
4. More specific email notifications were sent to users.

CARBLogger CentOS –

CARBDMS TEAM is currently working on the next version of CARBLogger. Earlier versions of CARBLogger have been based on a distribution of Linux called "Debian". While CARBLogger systems operating on the Debian platform were successful, CARBDMS TEAM encountered many challenges, as there was very little hardware support for the Debian OS.

To address these challenges, a collaborative effort is presently underway between CARBDMS TEAM and CARB's Office of Information Systems (OIS) to develop a new version of CARBLogger operating on the Community Enterprise Operating System (CentOS) distribution of Linux. CentOS is an open source enterprise version of Red Hat.