# Offshore CTD QC Process

1. Identify the latest data QC’d for a given station based on the latest date in the “\_qcd.csv” filename in Z:\CTD\_data\_repository\STATION, where “STATION” is the locator name.
   1. The latest QC’d date is 1/3/2023 in *ctd\_extract\_KSBP01\_10-20-1998-****01-03-2023****\_qcd.csv*
   2. If the filename wasn’t updated, open the file and look for the date of the latest data flag to verify. Always good to check this anyway. Sometimes Greg forgets to rename the file.
2. Download the latest data from <http://dnrlab-web:82/mwdataset.asp?staADD=24&UpDownVal=&rbUpDown=>
3. Append the latest data to the end of the “\_raw.csv” and “\_qcd.csv” files.
4. Save both, and rename the “\_raw.csv” file to include the latest date. Leave the “\_qcd.csv” file as-is for now.
5. Sanity check: make sure that the file lengths are identical for the “\_raw.csv” data and the “\_qcd.csv”.

**Note: at this point, check to see the how recently discrete bottle data have been uploaded/QCd. The visual QC scripts include bottle data in the generated figures if available, which is crucial for CTD data verification.**

1. Run the automated QC script (see below)
   1. Output = report of flagged profiles, .csv of the actual data points (1/sampling event, up+down)
2. Run the “QC\_plots\_monthly.R” and “QC\_plots\_multipanel.R” scripts on the raw data
   1. QC\_plots\_monthly.R: Change the value for the following variables:
      1. “stations” – can be a list of multiple locators
      2. “date\_hilite” – vector of the first date and latest date.
   2. QC\_plots\_multipanel.R
      1. “stations” – can be a list of multiple locators
      2. date\_begin – the first date that you would like to have figures for. Typically the date of the first profile that has not been QCd
      3. date\_end – the latest date. This can and exceed the latest date.
3. Visually inspect the figures for anomalies in data. AS A GROUP!!!
4. Mark any suspicious or bad data points from the multipanel figures in the “\_qcd.csv” file
   1. R = rejected, Q = questionable, blank = it’s fine
   2. Avoid TA (text available)
5. Run composite and stratfiication scripts to update the big files on the Z drive
6. Store QC information publicly – how? Socrata, direct or indirect write access to existing database, something else??

# Future Stuff

* Automated QC flags
  1. Flag all data when pressure/depth is negative
  2. Determine range for each parameter, per depth bin (of undermined size!), and identify if the data point is outside of that bin by an error margin
     + Two levels –
       - Hard-coded out-of-bounds values (e.g. anything outside is very likely “bad”) what are the cutoff values for each parameter?
       - From previous data, with a buffer what size buffer? +/- 2SD?
     + Error margin to be determined on a case-by-case basis for certain parameters, or based on standard deviation of the depth bin.
  3. Determine total change from prior and next data point. If this exceeds a predetermined amount for either (e.g. 1 mg/L for DO), flag the data point.
     + Moving average – take difference between value and average; is it beyond a certain %? What %?
     + Look up spike detection algorithms/functions in R – has someone already done the hard work?
     + Reference other related parameters for changes to determine if spikes are OK
       - Simultaneous changes in salinity and temperature in deep water
       - Chlorophyll fluorescence and dissolved oxygen at depth
  4. Comparison of downcast to upcast data - % change tolerance? Bin size?
  5. Comparison to bottle data. Determine average for depth bin (size TBD for each parameter), and check if it is within acceptable range of bottle sample.
  6. Programmatically identify weird stuff – how to quantify?
     + Out of water for multiple hours
     + Nonchanging data
     + Buoyancy frequency as a check
     + Large point to point variation – noisy data
     + Clogged plumbing (rapidly reducing salinity and DO)
     + Bubbles (rapidly reducing salinity, increasing DO)+
  7. **FAR FUTURE:** Machine learning based on previously QC’d data. Needs metadata.
* Database/IT needs
  1. A writeable database that can include data quality flags
  2. Consistent visualizations with Whidbey and Central Basin data
* Future Data flow
  1. SG collects CTD data.
  2. Uploads to somewhere. **Preliminary data is immediately available to the public, marked as “provisional” by default.**
  3. WQP is able to quickly scrape data from this provisional database (R, or otherwise) via an API. No spreadsheets involved.
  4. WQP runs automated script, and gets a report of all data flags.
  5. WQP runs data visualization script, and repeats the current QC process. Confirms and finalizes the QC data flags to apply.
     + Samples with QC flags contain the primary key for easy writing to the database
  6. WQP uploads updates to the database
  7. Data are immediately available, and “provisional” notice is removed from QC’d data.