**Status v1 (6/17/2021)**

**Who to contact if you have problems or comments**

This is an open source, collaborative effort and we welcome your feedback. If you encounter problems with the Shiny app and/or have suggestions on ways to further improve it, please contact Erik W. Leppo from Tetra Tech via email ([Erik.Leppo@tetratech.com](mailto:Erik.Leppo@tetratech.com)) or via a post on the ContDataQC GitHub page (<https://github.com/leppott/ContDataQC>).

**Experience to date**

We have been working on and off on the ContDataQC R package and Shiny app since 2017. Development of the ContDataQC R tools has been funded primarily by the United States Environmental Protection Agency (EPA) Office of Research and Development (ORD) as part of a larger project to establish Regional Monitoring Networks (RMNs) for freshwater streams and inland lakes. The lead EPA scientist on the RMN work is Britta Bierwagen ([bierwagen.britta@epa.gov](mailto:bierwagen.britta@epa.gov)). Shane Bowe from the Red Lake Band of Chippewa Indians has also funded parts of its development.

So far ContDataQC has only been used on continuous data that is being manually downloaded periodically (generally one to three times a year). It has not been used on ‘real-time’ continuous data. It was originally developed for sensors deployed in streams at single depths but is now being used on lake sensor array data as well, where sensors are being deployed at multiple, fixed depths at the same site. To date, we have worked primarily with thermal and hydrologic data (the primary variables being collected at RMN sites), but users have expressed interest in using it on other parameters as well. The Shiny app is currently set up to run dissolved oxygen, conductivity, pH, turbidity and chlorophyll-a data but the default QC test thresholds for those parameters will need to be carefully checked and refined.

**Plans for 2021-2022**

In fall of 2020, a major advancement occurred when we moved the Shiny app to the Tetra Tech Shiny server. This made the app much more accessible to RMN partners and to other outside users as well. EPA is planning to host the ContDataQC Shiny app on an EPA Shiny server as well. We expect this to happen over the coming year.

We aren’t planning to make any major changes to the main functions in ContDataQC but will continue to work on the items below as resources permit. We’ll also be shifting our focus more to data summary and visualization, for which we are developing separate R packages and Shiny apps (for more information, see bottom of page).

Here are the ContDataQC items we’re currently working on -

* Improving the instructional materials
* Refining the QC test thresholds, not just for temperature and hydrology (the primary variables being collected at RMN sites), but also for dissolved oxygen, conductivity, pH and turbidity.
* Gaining a better understanding of the most common QC issues being encountered and how to best detect and address (or potentially prevent) those issues.
* Improving guidance on addressing errors and flagged data points and documenting corrections and metadata.

ContDataQC ‘wish list’ items (that we lack funding to address but will tackle if resources permit) -

* Automated accuracy check
  + Users would upload in-situ files and the R code would automatically match the in-situ measurements with the closest sensor measurement, calculate the difference between the two measurements and report whether the sensor is meeting accuracy specs.
* Weather station check
  + Users would upload weather station data and the R code would match the measurements with the sensor data, generate time series plots in which the measurements are overlaid on one another, and generate some basic statistical measures to compare the two sets of data.
* Daily range check (max-min)
  + This QC check is used by the USFS NorWeST temperature sensor network. It would be added in as a new function.
* ‘Transition’ check
  + We’re finding most errors concentrated around the times when the sensors are being taken out of position for data download. If you’re only QCing individual files after each download, you’re not evaluating the transition between data from the previous file and the new file. This check would bring in data from 1-2 days prior to the start time of the file being evaluated and alert users to data points that need to be trimmed.
* Automatic adjustments to gross limit QC test thresholds based on season and depth
  + Right now the gross limit QC test is pretty limited because it doesn’t capture seasonal differences without the user creating multiple customized configuration files (and then the user would need to break up the input files for each time period). We’d like to figure out a way to have the R code automatically adjust the upper and lower gross limit thresholds based on seasonal considerations, and for lakes, for it to also make adjustments to gross limit thresholds based on depth.
* Drift correction
  + We’d add a feature that automates drift corrections.

**Other R packages and Shiny apps that may be of interest**

Tim Martin from Minnesota DNR ([tim.martin@state.mn.us](mailto:tim.martin@state.mn.us)) has been a great collaborator and has developed a QC app that includes the ContData QC flag tests as well as some other features (including an interactive plot in which data points are color-coded based on flags and the user can add or remove flags). The app is currently being hosted on the Tetra Tech Shiny server -

<https://tetratech-wtr-wne.shinyapps.io/logger_processing/>

If you use the app, Tim welcomes your feedback!

For summary and visualization of continuous lake data, we’re actively working on LakeMonitoR -

R package - <https://github.com/leppott/LakeMonitoR>

Shiny app - <https://tetratech-wtr-wne.shinyapps.io/LakeMonitoR/>

This work is being funded by Shane Bowe from the Red Lake Band of Chippewa Indians (via a BIA grant) and EPA ORD.

Over the coming year, we’ll also be working on new tools for summarizing and visualizing the stream RMN data (thermal, hydrologic and biological data).

There are other existing software programs for working with continuous data. One example is the open source Python-based tool by Horsburgh et al. (2015), which provides a data quality control work flow, allows for flagging of data based on visual inspection, can apply drift corrections and interpolate missing values (if desired) and saves both original and edited data for documentation. Other existing software options include a sensorQC R package by USGS (<https://github.com/USGS-R/sensorQC>) and rLakeAnalyzer (Read et al. 2011, Winslow et al. 2018), which have statistical outlier detection tools.