

# Introduction to file and data formats in rLakeAnalyzer

Luke Winslow

July 6, 2014

## 1 Introduction

This document is an introduction to handling the type of data typically used in rLakeAnalyzer. It will hopefully give the reader enough information to be able to quickly and effectively format your own data to take advantage of the more powerful features.

## 2 File Format

We have tried to use a simple but standard file format that eases import and parsing of data while still being easy to generate and edit using commonly available tools like Microsoft Excel. Below is a very simple example of how the files are structured.

| <b>datetime</b>  | <b>doobs_0.5</b> |
|------------------|------------------|
| 2008-07-01 01:00 | 8.3              |
| 2008-07-01 02:00 | 8.2              |
| 2008-07-01 03:00 | 8.2              |
| 2008-07-01 04:00 | 8.1              |

There are a few key aspects to these file structure to note. The date/time format, the format of the header, and the file naming scheme. These key points are discussed here.

### 2.1 DateTime Format

The date and time of all observations is stored in a single, string column. The header of this column must be the word "datetime" without quotes. It is also case insensitive so "DateTime" and other variations will work.

The datetime format itself is exclusively in an ISO-like format (ISO-8601). It is in most-to-least significant order. It requires a "-" (dash) delimited date format and a ":" (colon) delimited time. "yyyy-mm-dd HH:MM:SS". The date must come first and is separated from the time with a single space. Seconds are optional. This format can easily be created in Excel using a custom date/time format of "yyyy-mm-dd hh:mm:ss" without quotes.

Note: This format differs from the ISO-8601 format in that a space is used to separate the date and time. This is done to support the use of Microsoft Excel as Excel does not natively recognize the ISO format.

### 2.2 Header Format

The header is used to help identify both the variable type as well as the depth of observation of the data as well as distinguish the data columns from the datetime column. As mentioned above, a "datetime" column is required using the format described above.

The data columns must be identified with a variable type and optionally, a depth. For example, a water temperature collected at 1 meter depth would have the column header "wtr\_1". The usefulness of this simple format can be seen when dealing with profile data taken at many depths (see below).

| <b>datetime</b>  | <b>wtr_0.5</b> | <b>wtr_1</b> | <b>wtr_2</b> |
|------------------|----------------|--------------|--------------|
| 2008-07-01 01:00 | 22.3           | 22.3         | 21           |
| 2008-07-01 02:00 | 22.31          | 22.31        | 21           |
| 2008-07-01 03:00 | 22.31          | 22.31        | 21           |
| 2008-07-01 04:00 | 22.32          | 22.32        | 21           |

While any text can be used to describe a variable, the table below lists the current "standard" variables that are used by rLakeAnalyzer and other toolboxes for identifying commonly collected data in the most common units. If these standards are adhered to, many of the more helpful functions will work natively. For example, `water.density` expects temperature to be supplied in celsius, the default unit used for the "wtr" abbreviation.

| <b>Abbreviation</b> | <b>Variable</b>                | <b>Assumed Units</b> |
|---------------------|--------------------------------|----------------------|
| doobs               | Dissolved Oxygen Concentration | mg/L                 |
| wtr                 | Water Temperature              | °C                   |
| wnd                 | Wind Speed                     | m/s                  |
| airT                | Air Temperature                | °C                   |
| rh                  | Relative Humidity              | %                    |

### 2.3 File Format

The file format is a simple tab-delimited file. It is easy to export files of this format using Excel or even R itself. To export the appropriate format from R, use "write.table" as in the following example.

```
> tmp = data.frame()
> write.table(tmp, "test.wtr", sep='\t', row.names=FALSE)
```