

# NRT RAW Data Format

## Introduction

The O2A component INGEST uses so called *drivers* (actually simple data converters) to convert your sensor data to a common data format: NRT RAW. This is not yet the [NRT Data Format](#) but an intermediate format which all our tools of our INGEST-to-DASHBOARD workflow understand.

In order to make INGEST-to-DASHBOARD work easily for your data it would be very helpful for us that you provide a driver (script in Python, R, ...) able to handle your sensor data and create an NRT RAW file from that.

Afterwards we can easily map your sensor's data columns to actual SENSOR parameters.

## Example

```
datetime    TEMP [°C]    SAL [psu]
2019-02-28 15:50:00 -0.5445  34.1234
2019-02-28 15:50:01 -0.5444  34.1345
2019-02-28 15:50:02 -0.5552  34.1456
```

## Format

Plain-text table having columns for each parameter from your raw data files and rows for each individual time record.

A sample file is attached: [ferrybox\\_raw\\_SVL\\_20190219\\_230002\\_ferrybox-731101.nrtraw](#)

What	How
Character encoding	UTF-8
Field / column separator	Tabulator "\t"
End of record marker	New line "\n"
Header	<p>The header must consist of a column named "datetime" and 1-n columns having a static name describing your raw parameter. If you convert from a different table-like dataset your driver could also just take over the raw column headers. The "datetime" column must always be the first column. The column headers should not have any special characters except hyphen "-" or underscore "_" and no spaces.</p> <pre>datetime    TEMP    SAL</pre>
Header - Units	<p>Units may be given in square brackets [], following each column name, separated by a space " ":</p> <pre>datetime    TEMP [°C]    SAL [psu]</pre> <p>Units are optional but really make sense in this very first step because the INGEST-to-DASHBOARD workflow would automatically set up these units in databases such as the Dashboard.</p> <p>If you desire to store texts such as sample numbers or comments in your file, you have to append a unit [text]! Example:</p> <pre>datetime    TEMP [°C]    SAL [psu]    SAMPLE-ID [text] 2019-02-28 15:50:00    -0.5445    34.1234    PS83/1_233-1</pre>
Date time format	<p>Expected time zone: UTC</p> <p>Allowed date formats:</p> <pre>yyyy-mm-dd HH:MM:SS.fff yyyy-mm-dd HH:MM:SS yyyy-mm-ddTHH:MM:SS.fff yyyy-mm-ddTHH:MM:SS</pre>
Decimal symbol	Dot "."

Data values	<p>Allowed are decimal values (scientific format 1E-2 is supported), integer numbers or characters/texts/strings (just no tabulators!)</p> <p>34.56556    23    334.43E-2    PS83/1_233-1</p> <p>For use of Data-WS only: The Data-WS only accepts numbers as input for each column (including number formats such as 334.43E-2).</p> <p><u>About texts (or non-numeric characters):</u></p> <p>If you want to store texts, the column's unit must be [text]!  For columns which do not have the unit [text] all characters or words (such as "inf", "NaN", "SAMPLE1") will result in an import error.  Texts cannot contain tabs or newlines.</p>
Error values	Empty field (simply give a null string, but do not forget to add a tab as data field separator)
Missing values	Missing values do not need to be masked in a special way. The table cell just needs to be blank (no NA, null, etc.)