NRT Drivers

What is an NRT driver?

An NRT driver is a script/executable which converts a raw data file to a NRT RAW Data format.
NRT RAW is an intermediate file format our O2A-INGEST tools can handle to stream your data directly to the Dashboard and Data Webservice.
Check the list of drivers below - maybe there is already one matching your raw data?!
Or simply ask us if you want to provide your own driver. The requirements for that you can check below.

List of available drivers

Currently the following drivers are available to convert sensor's raw data to NRT RAW Data format:

Name	Sensor types	Language
adcpToNrt	RDI ADCP instruments (format pd0 data format such as contained in files .enc, .enx, ...)	Python3
ctdToNrt	Sea & Sun Technologies's CTD (format provided by STA acquisition software)	Python3
ferryboxToNrt	4h-Jena Ferrybox	Python3
sbe38ToNrt	Seabird SBE38 temperature probe (format provided by probe itself via serial line)	Python3
snowbuoyToNrt		Python3
toa5ToNrt		Python3

Provide your own driver

You may provide your own driver which should be able to convert your special raw data format to the NRT RAW Data format.

Requirements are:

- It needs to be executable on our Linux (Ubuntu 20.04) servers:
  - Python3 is currently or main driver's programming language.
  - Bash/Shell scripts would work (if you use AWK, SED, ...).
  - Binaries would work, if compiled for Linux (and if they do not bring to many weird dependencies).
  - R might work on request (not used yet)
- It should be callable simply by passing an input file path:
  - `<myscript> <inputfile>`
  - Example: `python3 /drivers/my-super-special-driver.py /isibhv/platforms/my/device/2020-10-16.dat`
- The standard output of your driver (stdout, all your print/echo/write commands do) is taken as the result and should fulfill the NRT RAW Data format.
  - 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
    ```
- Errors or warnings during your driver's runtime should be printed to `stderr` and NOT to `stdout`!
  - Every programming language provides output methods for that.
  - Only with that we can make sure that the data output and the non-data output is separated.
- As best practice your driver should return a proper exit code / exit status as most Linux tools do:
  - 0: Everything went fine.
  - 1-n: Something went wrong.
  - Only by that our NRT INGEST tools can recognize something went wrong.

We are here and happy to help, if you have any question on the drivers.