MeteoIO as a data publication service in the context of Arctic PASSION and WMO GCW

Mathias Bavay[1], Patrick Leibersperger[1], Øystein Godøy[2], Charles Fierz[1], Rodica Nitu[3]

[1] *WSL Institute for Snow and Avalanche Research SLF, Davos, Switzerland*
[2] *Norwegian Meteorological Institute, Oslo, Norway*
[3] *World Meteorological Organization, Geneva, Switzerland*

*This software has received funding from the World Meteorological Organization under grant agreement No. 29539/2022-1.9 as well as the European Union's Horizon 2020 research and innovation program under grant agreement No. 101003472 (**Arctic Passion**).*

Focus: research AWS

Large monitoring networks: lots of identical stations (WMO, etc)

Research AWS: stations installed to answer a specific scientific question

First Challenge: diversity

- Measured parameters, sampling rates
- Sources of data (database, webservice, files)
- Data formats (all variants of csv, …)
- Changes in the station setup at any time during the station’s lifetime

Example: Davos Stillberg station

```
1;2;3;4;5;6;7;8;9;                                          [97 more fields]
YEAR;MONTH;DAY;TIME;TEMP_ENGL_H;TEMP_ASP;TEMP_S100;TEMP_S50;TEMP_S10;  [97 more fields]
No;No;No;No;°C;°C;°C;°C;°C;                                 [97 more fields]
No;No;No;No;0.01;0.01;0.01;0.01;0.01;                       [97 more fields]
76;1;1;30;-177;-175;95;54;-10;                              [97 more fields]
76;1;1;130;-200;-202;94;53;-11;                             [97 more fields]
76;1;1;230;-186;-179;94;53;-11;                             [97 more fields]
76;1;1;330;-228;-235;94;53;-11;                             [97 more fields]
```

Sensors from multiple stations within 1 file

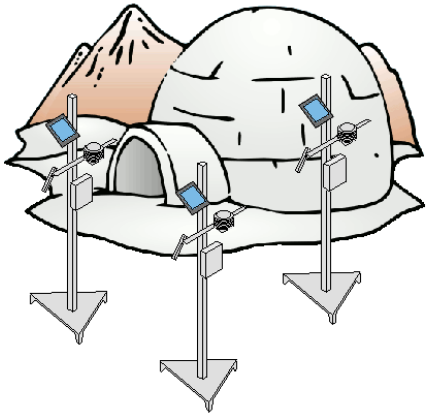
+ Changes in # of sensors within 1 file + sensors swapped over time (not matching the headers)

```
1;2;3;4;5;6;7;8;9;                                          [24 more fields]
1;2;3;4;No;No;5;6;30;                                       [24 more fields]
STAT_NR;YEAR;DAY_OF_YEAR;TIME;TEMP_VTP6;RH_VTP6;  [27 more fields]
No;No;No;No;°C;%;°C;%;mm;                                   [24 more fields]
No;No;No;No;1;1;1;1;1;                                      [24 more fields]
350;2000;272;920;9.46;64.14;-6999;-6999;0;                  [24 more fields]
350;2000;272;930;10.73;63;-64.1;-25;0;                      [24 more fields]
350;2000;272;940;9.92;59;-64.1;-25;0;    [24 more fields]
```

- AWS since 1975
- Lots of changes in types & number of sensors, different sampling rates
- Several full rebuild
- Nobody really looked into the old data

Ultimate goal

Research stations



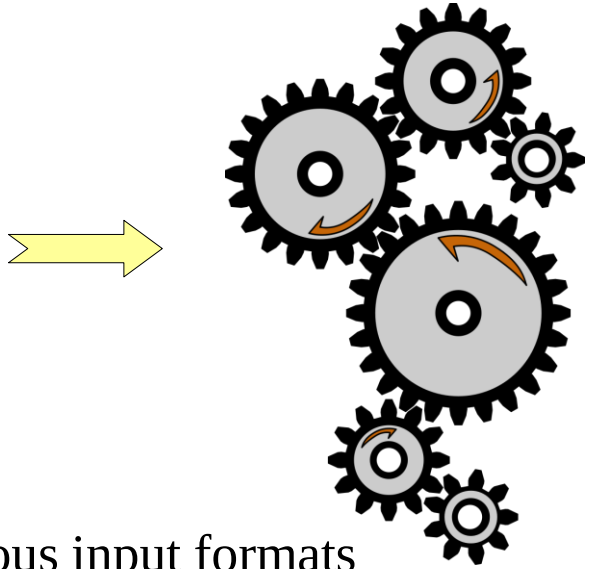
Findable

Accessible

Interoperable

Reusable

Requirement 1: standardization

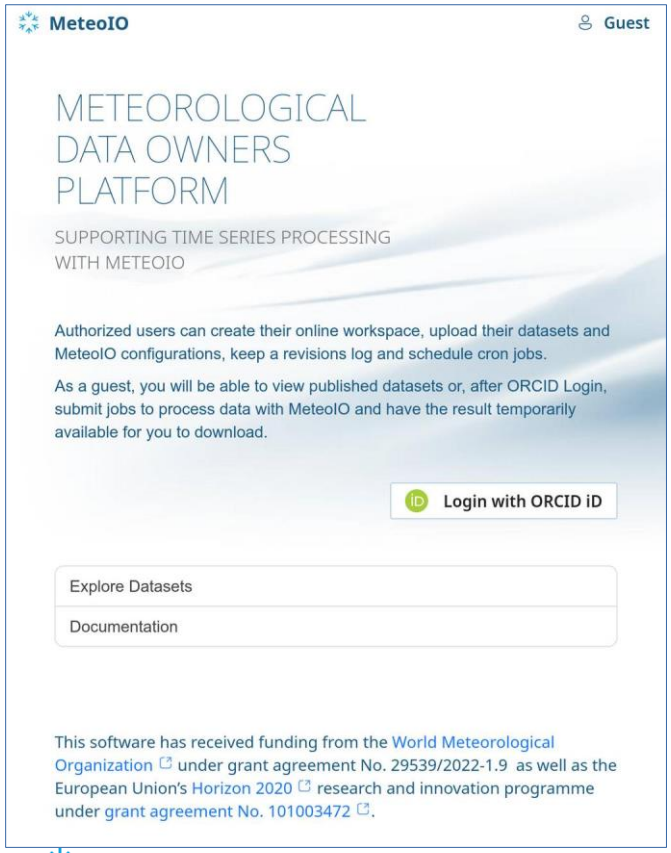


- Handle the various input formats
- Same output format
- Same parameter naming
- Same metadata standard

Requirement 2: reproducibility

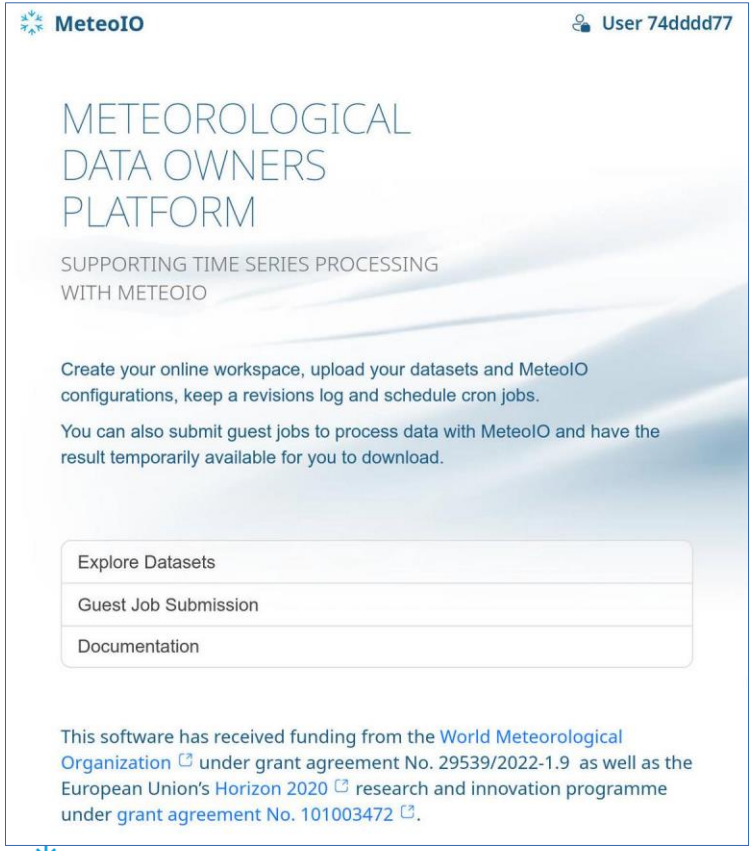
- Document ALL processing steps;
- no manual editing of data, generated on the fly;
- for any period of the station's lifetime!

Our new system: user experience



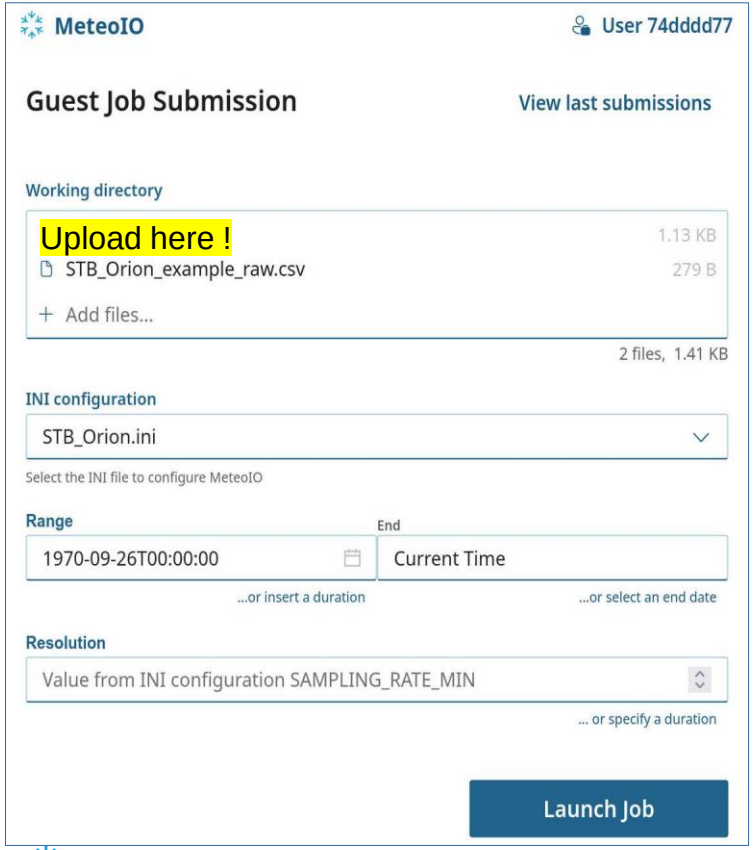
- Go to our web service home page
- Login with your Orcid ID

Our new system: user experience



Select «Guest Job Submission»

Our new system: user experience



- Upload data file
- Provide start and end time
- And launch job!

Our new system: user experie

MeteoIO

User 74dddd77

Job Result

View last submissions

ID: 018b6d96-4e7d-7efa-bf49-aff9ce15c400
Created: seconds ago
Status: Finished.
Wait time: 19ms
Processing time: 188ms
Disk usage: 46.82 KB

Output Files Logs

STB_Orion.nc 1.41 KB

And get a standardized file back!

```
netcdf STB_Orion {
dimensions:
  time = UNLIMITED ; // (3 currently)
variables:
  float time(time) ;
      time:standard_name = "time" ;
      time:units = "minutes since 1976-01-01 00:00:00" ;
      time:calendar = "gregorian" ;
      time:axis = "T" ;
  float orog ;
      orog:standard_name = "surface_altitude" ;
      orog:long_name = "height above mean sea level" ;
      orog:units = "m" ;
      orog:_FillValue = -999.f ;
      orog:positive = "up" ;
      orog:axis = "Z" ;
  float latitude ;
      latitude:standard_name = "latitude" ;
      latitude:units = "degree_north" ;
      latitude:_FillValue = -999.f ;
      latitude:axis = "Y" ;
  float longitude ;
      longitude:standard_name = "longitude" ;
      longitude:units = "degree_east" ;
      longitude:_FillValue = -999.f ;
      longitude:axis = "X" ;
  float slope ;
      slope:standard_name = "slope_angle" ;
      slope:long_name = "slope angle" ;
      slope:units = "degrees from horizontal" ;
      slope:_FillValue = -999.f ;
  float azimuth ;
      azimuth:standard_name = "slope_azi
      azimuth:long_name = "slope azimut
      azimuth:units = "degrees from nort
      azimuth:_FillValue = -999.f ;
```

Yes, there is a trick...

MeteoIO

User 74dddd77

Guest Job Submission

View last submissions

Working directory

Upload here !

STB_Orion_example_raw.csv 1.13 KB 279 B

+ Add files...

2 files, 1.41 KB

INI configuration

STB_Orion.ini

Select the INI file to configure MeteoIO

Range

1970-09-26T00:00:00

End

Current Time

...or insert a duration

...or select an end date

Resolution

Value from INI configuration SAMPLING_RATE_MIN

... or specify a duration

Launch Job

Yes, there is a trick...

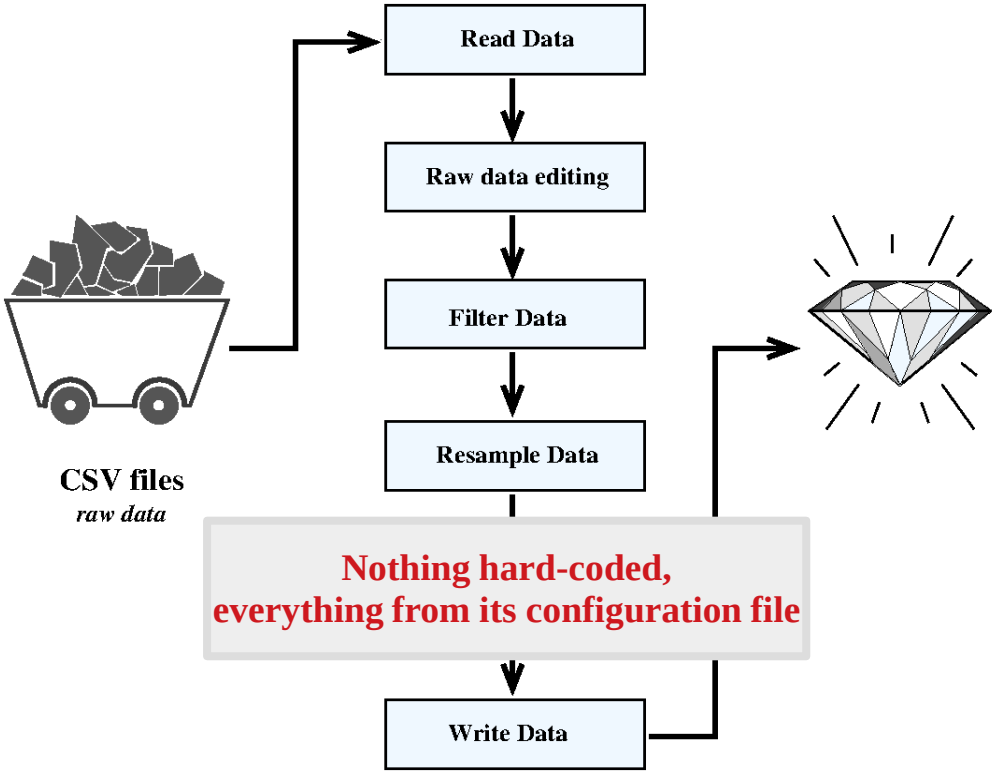
=> This single configuration file (per station) describes ALL the processing for the whole life of the station

Using the configuration file

Processing provided by our MeteoIO meteorological data pre-processing library:

- Started in late 2008
- In operational use for avalanche warning applications
- Open source

MeteoIO: principle of operation



- Fixed processing steps
- Many input formats to choose from (through plugins)
- Many editing / filters / corrections to choose from
- several output formats (through plugins)
- Nothing is hard-coded, everything comes from its config file
- Configuration file should be understandable by a human in 50+ years

MeteoIO: example configuration file

The INI file is structured in sections

```
[General]
BUFFER_SIZE = ${{370*3}}
BUFF_BEFORE = 1.5

[Input]
METEO = CSV
METEOPATH = input

COORDSYS = CH1903+
TIME_ZONE = 1.00
CSV_NAME = Weissfluhjoch
CSV_ID = WFJ_MET
POSITION = latlon (46.75, 9.80, 2200)

CSV_DELIMITER = ,
CSV_NR_HEADERS = 5
CSV_HEADER_REPEAT_MK = rawarchiver appending at
CSV_FIELDS = timestamp skip VW VW_MAX DW TA RH skip skip TSS
CSV_TIMESTAMP = COMBINED
CSV_DATETIME_SPEC = "YYYY-MM-DD HH24:MI:SS"

STATION1 = 2018/SCIENCE_5WFJ_MET_meteo.csv
STATION2 = 2019/SCIENCE_5WFJ_MET_meteo.csv
STATION3 = 2020/SCIENCE_5WFJ_MET_meteo.csv
STATION4 = 2021/SCIENCE_5WFJ_MET_meteo.csv

[Filters]
TA::FILTER1 = UNVENTILATED_T
TA::ARG1::TYPE = Huwald
TA::ARG1::SOIL_ALB = 0.80

[Output]
COORDSYS = CH1903
TIME_ZONE = 1.00

METEO = NETCDF
METEOPATH = output
NETCDF_SCHEMA_METEO = CF-1.6
ACDD_METADATA = TRUE
ACDD_SUMMARY = Research meteo Station at Davos Weissfluhjoch
WIGOS_ID = 0-20000-0-06780
ACDD_CREATOR = Mathias Bavay
ACDD_LICENSE = CC-BY-NC

SAMPLING_RATE_MIN = 10
```

MeteoIO: example configuration file

```
[General]
BUFFER_SIZE = ${{370*3}}
BUFF_BEFORE = 1.5

[Input]
METEO = CSV
METEOPATH = input

COORDSYS = CH1903+
TIME_ZONE = 1.00
CSV_NAME = Weissfluhjoch
CSV_ID = WFJ_MET
POSITION = latlon (46.75, 9.80, 2200)

CSV_DELIMITER = ,
CSV_NR_HEADERS = 5
CSV_HEADER_REPEAT_MK = rawarchiver appending at
CSV_FIELDS = timestamp skip VW VW_MAX DW TA RH skip skip TSS
CSV_TIMESTAMP = COMBINED
CSV_DATETIME_SPEC = "YYYY-MM-DD HH24:MI:SS"

STATION1 = 2018/SCIENCE_5WFJ_MET_meteo.csv
STATION2 = 2019/SCIENCE_5WFJ_MET_meteo.csv
STATION3 = 2020/SCIENCE_5WFJ_MET_meteo.csv
STATION4 = 2021/SCIENCE_5WFJ_MET_meteo.csv

[Filters]
TA::FILTER1 = UNVENTILATED_T
TA::ARG1::TYPE = Huwald
TA::ARG1::SOIL_ALB = 0.80

[Output]
COORDSYS = CH1903
TIME_ZONE = 1.00

METEO = NETCDF
METEOPATH = output
NETCDF_SCHEMA_METEO = CF-1.6
ACDD_METADATA = TRUE
ACDD_SUMMARY = Research meteo Station at Davos Weissfluhjoch
WIGOS_ID = 0-20000-0-06780
ACDD_CREATOR = Mathias Bavay
ACDD_LICENSE = CC-BY-NC

SAMPLING_RATE_MIN = 10
```

File reading plugin & location

MeteoIO: example configuration file

```
[General]
BUFFER_SIZE = ${{370*3}}
BUFF_BEFORE = 1.5

[Input]
METEO = CSV
METEOPATH = input

COORDSYS = CH1903+
TIME_ZONE = 1.00
CSV_NAME = Weissfluhjoch
CSV_ID = WFJ_MET
POSITION = latlon (46.75, 9.80, 2200)

CSV_DELIMITER = ,
CSV_NR_HEADERS = 5
CSV_HEADER_REPEAT_MK = rawarchiver appending at
CSV_FIELDS = timestamp skip VW VW_MAX DW TA RH skip skip TSS
CSV_TIMESTAMP = COMBINED
CSV_DATETIME_SPEC = "YYYY-MM-DD HH24:MI:SS"

STATION1 = 2018/SCIENCE_5WFJ_MET_meteo.csv
STATION2 = 2019/SCIENCE_5WFJ_MET_meteo.csv
STATION3 = 2020/SCIENCE_5WFJ_MET_meteo.csv
STATION4 = 2021/SCIENCE_5WFJ_MET_meteo.csv

[Filters]
TA::FILTER1 = UNVENTILATED_T
TA::ARG1::TYPE = Huwald
TA::ARG1::SOIL_ALB = 0.80

[Output]
COORDSYS = CH1903
TIME_ZONE = 1.00

METEO = NETCDF
METEOPATH = output
NETCDF_SCHEMA_METEO = CF-1.6
ACDD_METADATA = TRUE
ACDD_SUMMARY = Research meteo Station at Davos Weissfluhjoch
WIGOS_ID = 0-20000-0-06780
ACDD_CREATOR = Mathias Bavay
ACDD_LICENSE = CC-BY-NC

SAMPLING_RATE_MIN = 10
```

Basic geographic metadata

MeteoIO: example configuration file

```
[General]
BUFFER_SIZE = ${{370*3}}
BUFF_BEFORE = 1.5

[Input]
METEO = CSV
METEOPATH = input

COORDSYS = CH1903+
TIME_ZONE = 1.00
CSV_NAME = Weissfluhjoch
CSV_ID = WFJ_MET
POSITION = latlon (46.75, 9.80, 2200)

CSV_DELIMITER = ,
CSV_NR_HEADERS = 5
CSV_HEADER_REPEAT_MK = rawarchiver appending at
CSV_FIELDS = timestamp skip VW VW_MAX DW TA RH skip skip TSS
CSV_TIMESTAMP = COMBINED
CSV_DATETIME_SPEC = "YYYY-MM-DD HH24:MI:SS"

STATION1 = 2018/SCIENCE_5WFJ_MET_meteo.csv
STATION2 = 2019/SCIENCE_5WFJ_MET_meteo.csv
STATION3 = 2020/SCIENCE_5WFJ_MET_meteo.csv
STATION4 = 2021/SCIENCE_5WFJ_MET_meteo.csv

[Filters]
TA::FILTER1 = UNVENTILATED_T
TA::ARG1::TYPE = Huwald
TA::ARG1::SOIL_ALB = 0.80

[Output]
COORDSYS = CH1903
TIME_ZONE = 1.00

METEO = NETCDF
METEOPATH = output
NETCDF_SCHEMA_METEO = CF-1.6
ACDD_METADATA = TRUE
ACDD_SUMMARY = Research meteo Station at Davos Weissfluhjoch
WIGOS_ID = 0-20000-0-06780
ACDD_CREATOR = Mathias Bavay
ACDD_LICENSE = CC-BY-NC

SAMPLING_RATE_MIN = 10
```

Description of the files structure

MeteoIO: example configuration file

```
[General]
BUFFER_SIZE = ${{370*3}}
BUFF_BEFORE = 1.5

[Input]
METEO = CSV
METEOPATH = input

COORDSYS = CH1903+
TIME_ZONE = 1.00
CSV_NAME = Weissfluhjoch
CSV_ID = WFJ_MET
POSITION = latlon (46.75, 9.80, 2200)

CSV_DELIMITER = ,
CSV_NR_HEADERS = 5
CSV_HEADER_REPEAT_MK = rawarchiver appending at
CSV_FIELDS = timestamp skip VW VW_MAX DW TA RH skip skip TSS
CSV_TIMESTAMP = COMBINED
CSV_DATETIME_SPEC = "YYYY-MM-DD HH24:MI:SS"

STATION1 = 2018/SCIENCE_5WFJ_MET_meteo.csv
STATION2 = 2019/SCIENCE_5WFJ_MET_meteo.csv
STATION3 = 2020/SCIENCE_5WFJ_MET_meteo.csv
STATION4 = 2021/SCIENCE_5WFJ_MET_meteo.csv

[Filters]
TA::FILTER1 = UNVENTILATED_T
TA::ARG1::TYPE = Huwald
TA::ARG1::SOIL_ALB = 0.80

[Output]
COORDSYS = CH1903
TIME_ZONE = 1.00

METEO = NETCDF
METEOPATH = output
NETCDF_SCHEMA_METEO = CF-1.6
ACDD_METADATA = TRUE
ACDD_SUMMARY = Research meteo Station at Davos Weissfluhjoch
WIGOS_ID = 0-20000-0-06780
ACDD_CREATOR = Mathias Bavay
ACDD_LICENSE = CC-BY-NC

SAMPLING_RATE_MIN = 10
```

All files that are read by this dataset (per file config keys are supported)

MeteoIO: example configuration file

```
[General]
BUFFER_SIZE = ${{370*3}}
BUFF_BEFORE = 1.5

[Input]
METEO = CSV
METEOPATH = input

COORDSYS = CH1903+
TIME_ZONE = 1.00
CSV_NAME = Weissfluhjoch
CSV_ID = WFJ_MET
POSITION = latlon (46.75, 9.80, 2200)

CSV_DELIMITER = ,
CSV_NR_HEADERS = 5
CSV_HEADER_REPEAT_MK = rawarchiver appending at
CSV_FIELDS = timestamp skip VW VW_MAX DW TA RH skip skip TSS
CSV_TIMESTAMP = COMBINED
CSV_DATETIME_SPEC = "YYYY-MM-DD HH24:MI:SS"

STATION1 = 2018/SCIENCE_5WFJ_MET_meteo.csv
STATION2 = 2019/SCIENCE_5WFJ_MET_meteo.csv
STATION3 = 2020/SCIENCE_5WFJ_MET_meteo.csv
STATION4 = 2021/SCIENCE_5WFJ_MET_meteo.csv

[Filters]
TA::FILTER1 = UNVENTILATED_T
TA::ARG1::TYPE = Huwald
TA::ARG1::SOIL_ALB = 0.80

[Output]
COORDSYS = CH1903
TIME_ZONE = 1.00

METEO = NETCDF
METEOPATH = output
NETCDF_SCHEMA_METEO = CF-1.6
ACDD_METADATA = TRUE
ACDD_SUMMARY = Research meteo Station at Davos Weissfluhjoch
WIGOS_ID = 0-20000-0-06780
ACDD_CREATOR = Mathias Bavay
ACDD_LICENSE = CC-BY-NC

SAMPLING_RATE_MIN = 10
```

Applying some corrections on the data

MeteoIO: example configuration file

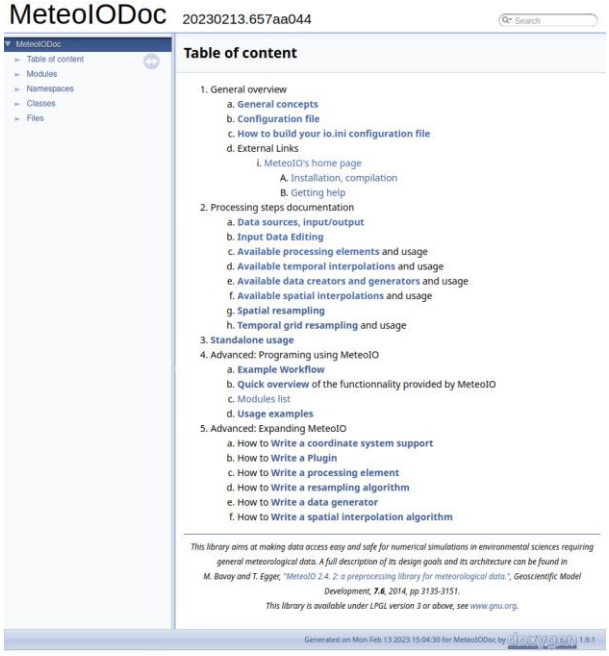
```
[General]
BUFFER_SIZE = ${{370*3}}
BUFF_BEFORE = 1.5

[Input]
METEO = CSV
METEOPATH = input

COORDSYS = CH1903+
TIME_ZONE = 1.00
CSV_NAME = Weissfluhjoch
CSV_ID = WFJ_MET
POSITION = latlon (46.75, 9.80, 2200)

CSV_DELIMITER = ,
CSV_NR_HEADERS = 5
CSV_HEADER_REPEAT_MK = rawarchiver appending at
CSV_FIELDS = timestamp skip VW VW_MAX DW TA RH skip skip TSS
CSV_TIMESTAMP = COMBINED
CSV_DATETIME_SPEC = "YYYY-MM-DD HH24:MI:SS"

STATION1 = 2018/SCIENCE_5WFJ_MET_meteo.csv
STATION2 = 2019/SCIENCE_5WFJ_MET_meteo.csv
STATION3 = 2020/SCIENCE_5WFJ_MET_meteo.csv
STATION4 = 2021/SCIENCE_5WFJ_MET_meteo.csv

[Filters]
TA::FILTER1 = UNVENTILATED_T
TA::ARG1::TYPE = Huwald
TA::ARG1::SOIL_ALB = 0.80

[Output]
COORDSYS = CH1903
TIME_ZONE = 1.00

METEO = NETCDF
METEOPATH = output
NETCDF_SCHEMA_METEO = CF-1.6
ACDD_METADATA = TRUE
ACDD_SUMMARY = Research meteo Station at Davos Weissfluhjoch
WIGOS_ID = 0-20000-0-06780
ACDD_CREATOR = Mathias Bavay
ACDD_LICENSE = CC-BY-NC

SAMPLING_RATE_MIN = 10
```

Output format & metadata to be added

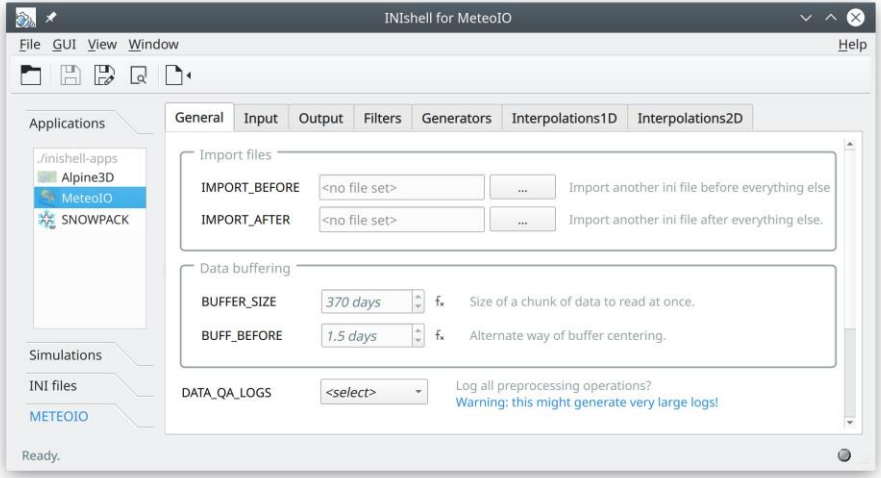
Writing INI files

Writing INI files is key to the system

Extensive online documentation



Dynamically generated GUI, direct links to the online documentation



"Inishell 2.0: semantically driven automatic GUI generation for scientific models", Bavay et al., 2022, gmd
https://doi.org/10.5194/gmd-15-365-2022

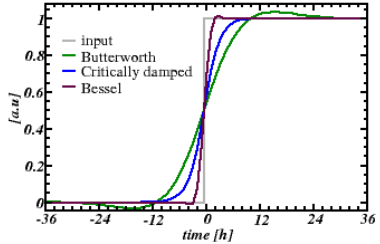
MeteoIO: Data QA/QC

- Currently, 41 filters to choose from
- Stack as many filters as you want, per parameter
- Extensive documentation

Infinite Impulse Response (IIR) filter.

This filter can either be used as a low pass or high pass filter. It is based on a Critically Damped, 2 poles filter (considering that it is better avoid overshooting even at the cost of a gentler falloff). It takes the following arguments:

- FREQ_RESPONSE: frequency response, either LP (for *Low Pass*) or HP (for *High Pass*);
- CUTOFF: The cutoff **period** (defined as the frequency at a -3dB gain) given in seconds;
- TYPE: either CRITICALLY_DAMPED (default), BUTTERWORTH or BESSEL (see figure below);
- SINGLE_PASS: Normally, the phase is removed by bidirectional filtering, ie. running the filter twice, first backward and then forward (this also squares the amplitude response). If set to TRUE, this bidirectional filtering is disabled.



Infinite Impulse Response filter: step response, LP bidirectional filtering over 24 hours

```
HS::filter1      = IIR
HS::arg1::freq_response   = LP
HS::arg1::type    = CRITICALLY_DAMPED
HS::arg1::cutoff = 10800 ;ie. 3 hours
```

To know more: http://unicorn.us.com/trading/allpolefilters.html and http://www.dspguide.com/ch19/4.htm.

MeteoIO: Data QA/QC

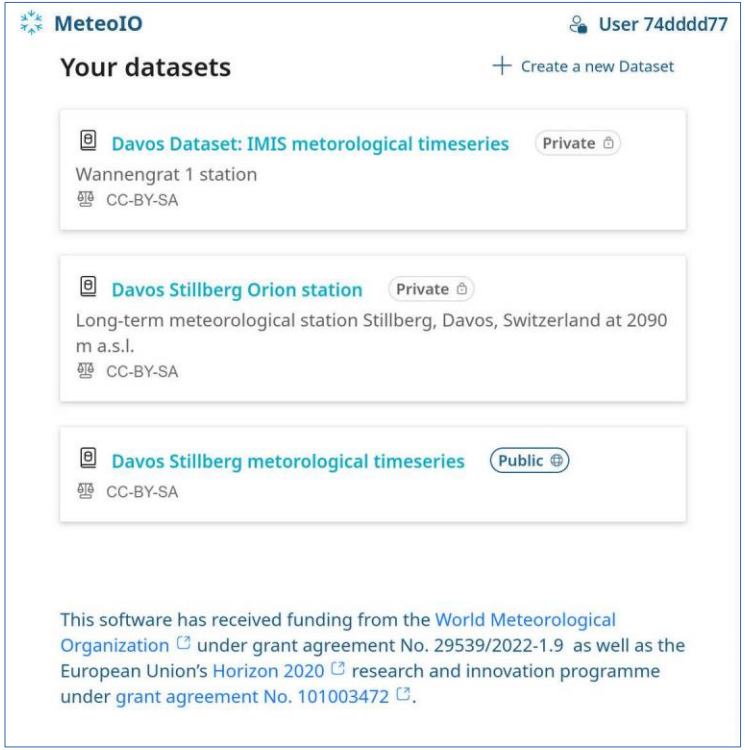
Enable the “DATA_QA” option, then request some data...

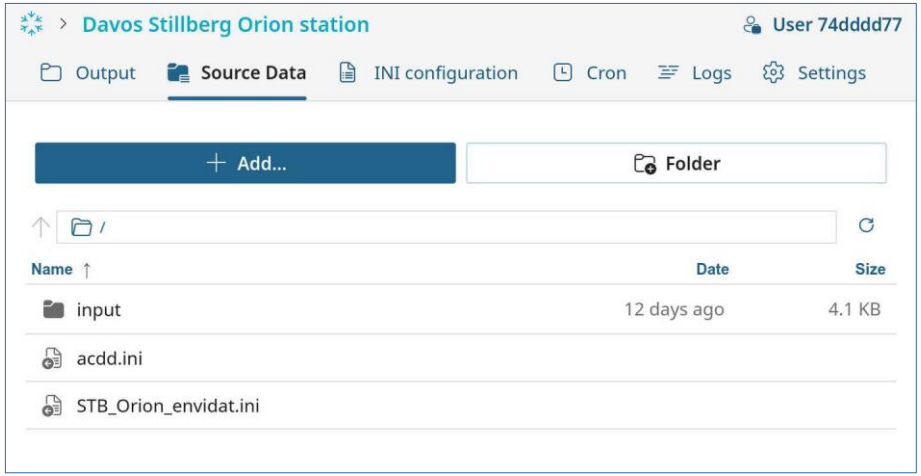
Filtering	ARO2::HS::MAD	2018-09-30T22:00:00
Filtering	SIM2::HS::MIN	2018-09-25T14:00:00
Resampling	ALI2::RH::LINEAR	2018-09-27T06:00:00
Resampling	PAR2::RSWR::LINEAR	2018-09-27T06:00:00
Missing	DAV5::TSG	2018-09-27T06:00:00
Missing	ELS2::RH	2018-09-27T06:00:00

Data Quality Tool

Station ▲▼	PSUM ▲▼	RSWR ▲▼	TA ▲▼	TSG ▲▼	SUM ▲▼
BER3	128	128	128	128	512
SCH2	1			336	337
OBW3	47				47
DIA2					0
CAM3	...			...	0
ATT2					0

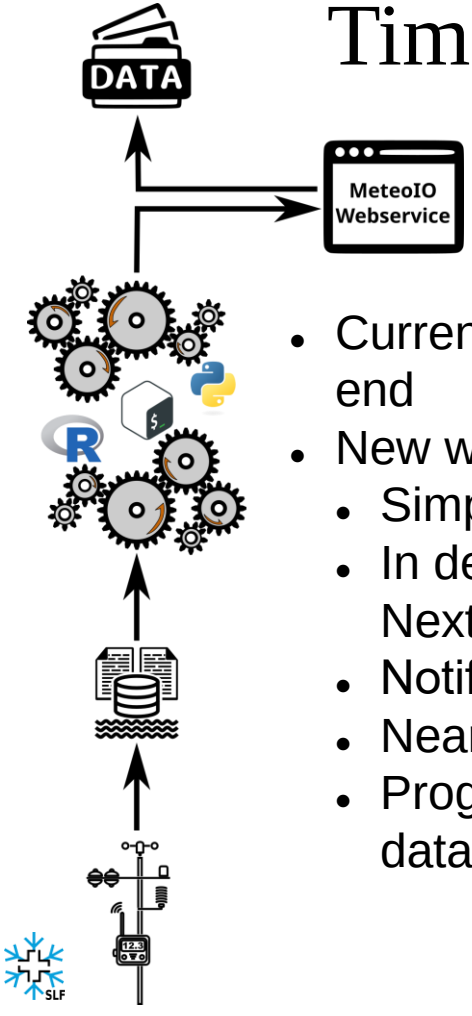
Not only guest, but as Data Owner



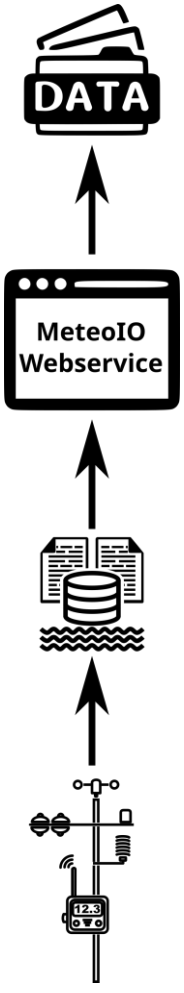


- Datasets permanently on the system
- Provide source data & INI file
- Automation with sftp + cronjob
- Anonymous users can see your public datasets
- Deliver up-to-date data as well as on-demand

Time to re-think our workflow!



- Current way, as presented: MeteoIO on the side, in the end
- New way: MeteoIO from the start
  - Simpler workflow
  - In dev: getting data from shared drives, Amazon S3, Nextcloud
  - Notifications from data QA→ quick reaction
  - Near-real time data delivery to operational users
  - Progressive data refining, snapshot to publish research dataset



New workflow in real life

Real life:

- Very little time to care for AWS data in research…
- Often neglected

New workflow:

- Empower Data Owners;
- Make production of data more robust;
- Increase data quality;
- **As a side effect**: data is standardized and FAIR, and it comes for free!

More information

- Data processor: MeteoIO pre-processing library, see (Bavay & Egger, 2014, gmd) https://doi.org/10.5194/gmd-7-3135-2014

- GUI for configuration file: Inishell see (Bavay et al., 2022, gmd) https://doi.org/10.5194/gmd-15-365-2022

- Earlier version for the WMO Global Cryosphere Watch, see (Bavay et al., 2020, dsj) http://doi.org/10.5334/dsj-2020-006