PAUL SCHERRER INSTITUT PSI

AWI Department Meeting – 25th July 2023

Overview:

- Dima – Identity management
- Christian – Jupyter widgets
- Klaus – BEC at the BLs
- Sven – SwissFEL Tools

Sven Augustin :: SCD/AWI/EIDO (7901) :: Paul Scherrer Institut

Tools at SwissFEL (cont'd): stand & grum

AWI Department Meeting – 25th July 2023

stand

stand – Motivation

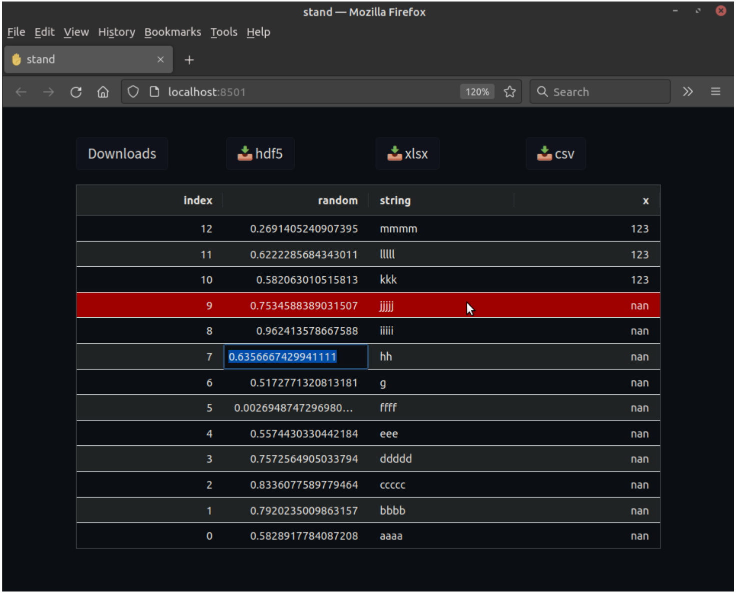
- General demand for customizable “run table”

- Use case:
  - Gather information about each recording/scan in tabular form
  - Allow commenting on runs after the recording
  - Allow manual data entry for devices that are not integrated

- Constraints / Wishes:
  - Should **not** use Google Spreadsheets, etc.
  - Should be accessible for data analysis (i.e., filtering) and Excel, etc.
  - Should be visible and editable from all consoles
  - (Should use off-the-shelf components / be simple to make)

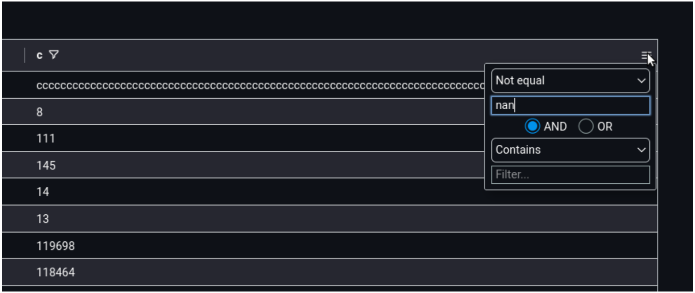
PAUL SCHERRER INSTITUT PSI

stand – Overview



index	random	string	x
12	0.2691405240907395	mmmm	123
11	0.6222285684343011	lllll	123
10	0.582063010515813	kkk	123
9	0.7534588389031507	jjjjj	nan
8	0.962413578667588	iiiii	nan
7	0.6356667429941111	hh	nan
6	0.5172771320813181	g	nan
5	0.0026948747296980…	ffff	nan
4	0.5574430330442184	eee	nan
3	0.7572564905033794	ddddd	nan
2	0.8336077589779464	ccccc	nan
1	0.7920235009863157	bbbb	nan
0	0.5828917784087208	aaaa	nan

stand – Overview (Filtering & Sorting)



c
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
8
111
145
14
13
119698
118464

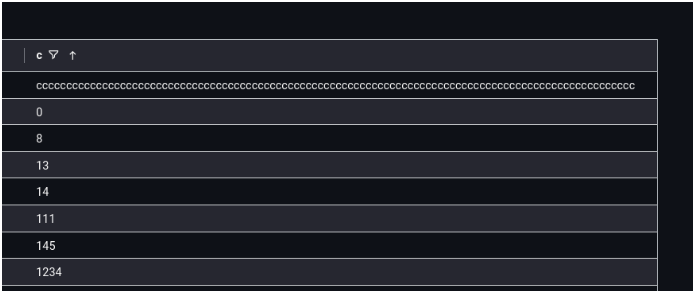
Not equal

nan

● AND ○ OR

Contains

Filter...



c ↑
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
0
8
13
14
111
145
1234

stand – Overview

- Ingredients:
  - Streamlit webapp
  - Ag-Grid spreadsheet component
  - CherryPy REST-API
  - Pandas DataFrames
  - A bit of glue Python code ...

- Simple client code:

```
from client import Client

c = Client()

c.add_row(a=1, b=2.3, c="four")
```

- Loading DataFrame in an analysis:

```
import pandas as pd

df = pd.read_hdf("output.h5")
```

grum

(**G**UI for **R**emote **U**nified **M**onitoring)

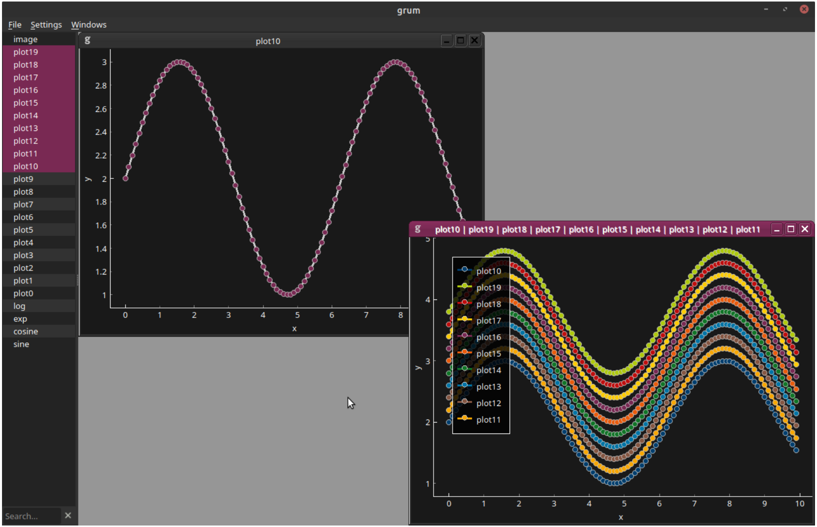
grum – Motivation

- Live visualization (and light-weight analysis*)

- Use cases:
  - Live plotting scans
  - Comparing current and previous scans of the same type
  - Live monitoring

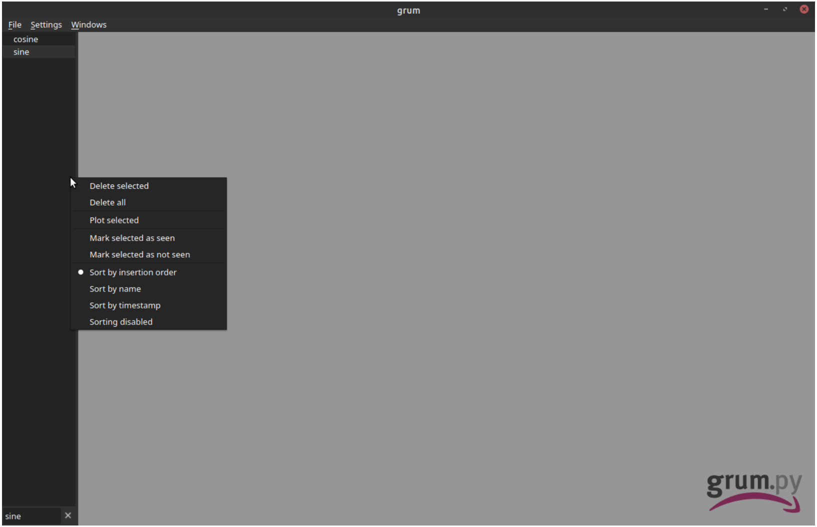
*I won't show this part this time, happens in slic `Sensors`.

grum – Overview



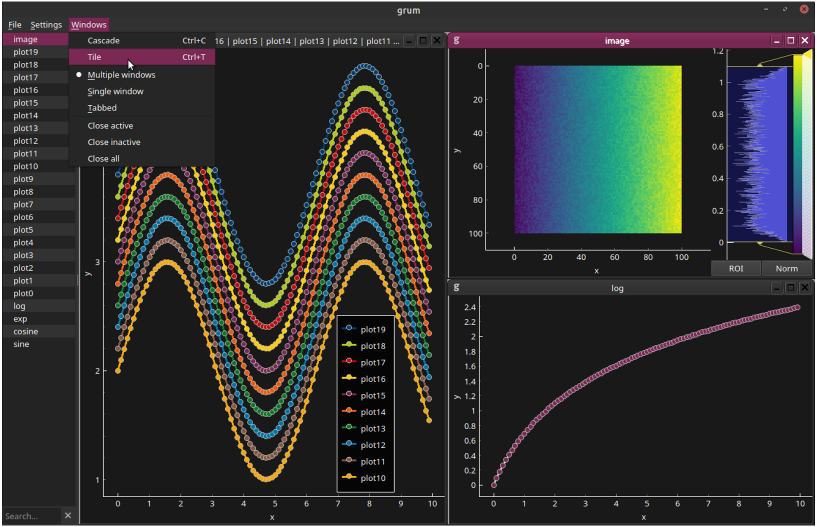
Single & Overlay Plot (both update in real time)

grum – Overview



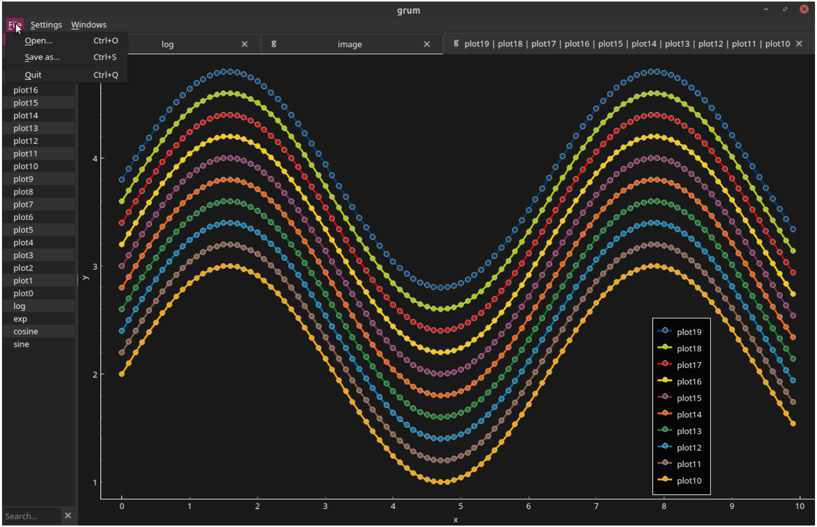
Searching / Filtering / Sorting / Deleting

grum – Overview



Tiling etc.

grum – Overview



Tabbed Mode / Opening & Saving hdf5 Files

grum – Overview

- Ingredients:
    - GUI via pyqt
    - Plots via pyqtgraph
    - RPC server via xmlrpc

- Simple client code:

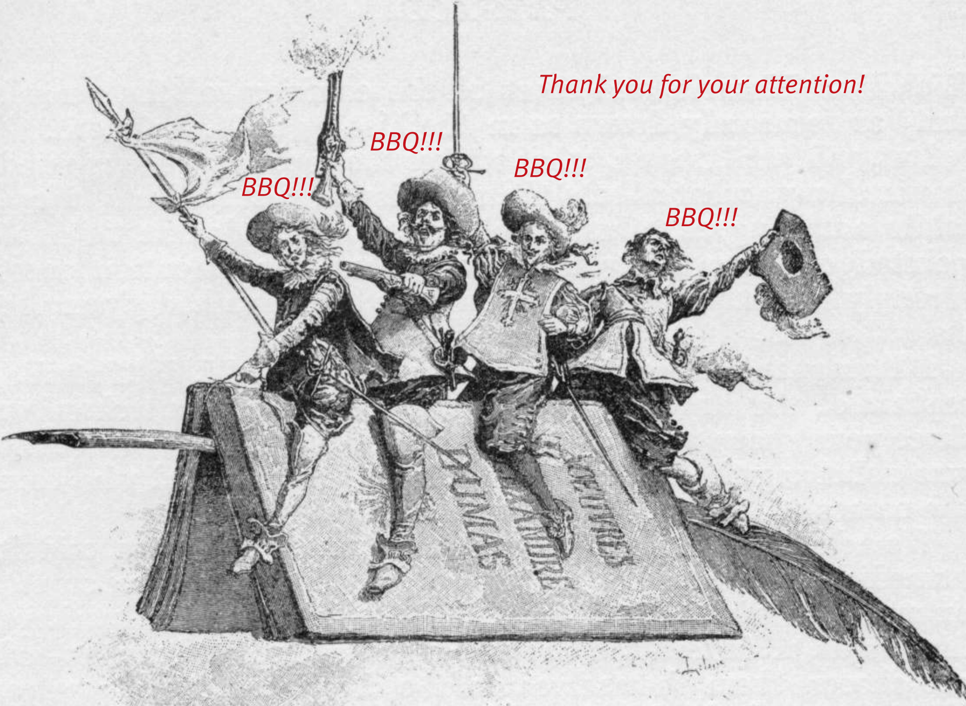
```
from stand.client import Client

c = Client()

cfg = {
    "xlabel": adjustable.name,
    "ylabel": sensor.name
}

c.new_plot(scan_name, cfg)

for step in scan:
    x = adjustable.get()
    y = sensor.get()
    c.append_data(scan_name, (x, y))
```



Thank you for your attention!