

## 1.2.2 Details of the area detector analysis: DetObjectFunc

This section is aimed at explaining the way the analysis of area detector works. It is intended for more expert use cases, since working with the producer and the producer template is enough for most experiments.

- [Intro to DetObjectFunc](#)
- [Make my own analysis function](#)

### Intro to DetObjectFunc

Area detector analysis in smalldata\_tools is handled using subclasses of DetObjectFunc. In order to be integrated in the workflow, the function must subclass that object.

Let's start with an example of how to add a ROI function to a detector and process it over a couple events:

#### DetObjectFunc simple example

```
import psana

from smalldata_tools.DetObject import DetObject, DetObjectFunc
from smalldata_tools.SmallDataUtils import getUserData
from smalldata_tools.ana_funcs.roi_rebin import ROIFunc

# get psana data source
exp = 'xpptut15'
run = 650
dsname = f'{exp}:{run}'
ds = psana.MPIDataSource(dsname)

# instantiate Jungfrau detector object
det = DetObject('jungfraulM', ds.env(), int(run))

# prepare the ROI function
# See https://github.com/slac-lcls/smalldata_tools/blob/master/smalldata_tools/ana_funcs/roi_rebin.py
# for info on the ROI function arguments
func_kwargs = {}
func_kwargs['ROI'] = [ [[0,1],[100,120],[50,60]] ] # a random ROI
func_kwargs['thresADU'] = None
func_kwargs['writeArea'] = False
func = ROIFunc(**func_kwargs)

# Attach function to the detector
det.addFunc(func)

# Process a couple events
userDict = {} # dictionary to store detector data

small_data = ds.small_data('./test.h5', gather_interval=5) # file to save data to
ds.break_after(5) # stop event loop after 5 events

for nevt,evt in enumerate(ds.events()): # usual psana event loop
    det.getData(evt) # get the detector data
    det.processFuncs() # process the attached functions
    userDict[det._name]=getUserData(det) # get the function results

    small_data.event(userDict) # write data to h5
```

As seen from this simple example, the workflow is as follow:

- Instantiate the detectors
- Setup the analysis functions with all its kwargs
- use `det.addFunc(...)` to add a function to the pipeline. Multiple functions can be added.  
It can happen that certain function modify the detector data directly and will lead to unexpected behavior or errors. Please consult with us if you want to use multiple functions on the same detector.
- `DetObjectFunc` also has a `addFunc` method, so that function can be daisy-chained. This is also contingent to the output of the first function being a compatible input to the second one, which is not always the case. For example, one can chain a ROI function to a projection to get the projection of the ROI only. Again, please consult with us for this.
- Run the psana event loop, calling the relevant smalldata\_tools methods and functions to process the detector according to the pipeline.

## Make my own analysis function

TO DO