# DataViewerBase

**DataViewerBase** package provides GUI and tools for checking images measured in real time.  
It was created to accumulate the data measured by the camera at SACLA in SPring-8 and check them.

## System requirements

The following packages and versions are required to use DataViewerBase.  
Among these, olpy and stpy are modules used on SACLA's HPC.  
It is not necessary if you only check the operation.

* Python ≥ 3
* PyQt4 (pyqt ≥ 4.11)
* pyqtgraph ≥ 0.10.0
* pyzmq ≥ 16.0
* olpy
* dbpy

## how to use

1. Open two terminals (command prompt). For convenience, we will call them A and B.
2. On terminal A, move to the DataViewerBase directory (folder).
   * This will start retrieving data from the database.
3. Execute python getDatawithOLPY.py.
4. On Terminal B, move to the DataViewerBase directory (folder).
5. Execute python main.py.
   * This will launch the GUI.

### Functions of DataViewerBase

* By pressing the Start button, data accumulation and display will start.
* Clear data by pressing Clear button.
* Save data at that point by pressing the Save button.
* By pressing the Window button, you can check the data in another window.
  + &8251; This is a function not particularly necessary on 2017/10/15.

### Problems found on 2017/10/15

The reference tag can not be accurately judged at the time of Run switching and therefore the signal and BG may not be correctly discriminated in some cases. Please restart getDatawithOLPY.py when such symptoms appear.

## Settings

Port and other settings are done with two files.

### /config\_getdata.json

This is the setting to use with getDatawithOLPY.py.

# /config\_getdata.json  
  
\* port : port numbrt where each data is sent  
 + sig\_wl : for signal with laser  
 + sig\_wol : for signal without laser  
 + bg\_wl : for BG with laser  
 + bg\_wol : for BG without laser  
\* port\_info : port number for miscellaneous information  
\* interval : interval between sending data (sec)  
\* timeout : period of work of getDatawithOLPY.py  
\* GetDataClass : parameters for GetDataClass  
 + detId : ID of detector  
 + channel : channel (currently unused)  
 + cycle : # of tags in one cycle (or measurement)  
 + bl : beamline number  
 + limNumImg : limit of # of images obtained at once  
\* signal\_flag : index of each data type in one cycle

### /gui/config.json

This is the setting used with DataViewerBase.py.

# /gui/config.json  
  
\* online : true = online mode  
\* closing\_dialog : true = show a dialog when closing the GUI  
\* currentDir : current directory  
\* emulate : true = emulate mode (currently unused)  
\* font\_size\_button : font size of buttons  
\* font\_size\_label : font size of labels  
\* font\_size\_groupbox\_title : font size of title of each groupbox

## Constitution

The structure of the DataViewerBase package is shown in the following figure.  
The function to retrieve data from the SACLA database is done by the getDatawithOLPY.py script.  
Data is displayed by the DataViewerBase class (GUI).

DataViewerBase/  
 - anatools/ : analysis tools for VMI  
 - core/ : core modules  
 + decorator.py  
 Some decorators.  
 + GetDataClass.py  
 Classes used to get data from the database.  
 + Worker.py  
 Worker classes.  
 currently GetDataWorker3 is used.  
 + ZeroMQ.py  
 Publisher/listener classes using ZeroMQ.  
 - gui/ : GUI classes  
 + AnalyzeWindow.py  
 Window class for analysis.  
 (under construction)  
 + DataViewerBase.py  
 The main class of this package.  
 + PlotWindow.py  
 GUI class for showing a data image and some graphs.  
 + getDatawithOLPY.py  
 a script which starts to get and send data.  
 + main.py  
 a script which starts DataViewerBase.

## Data flow

Firstly getDatawithOLPY.py acquires data accumulated in SACLA's server.  
They are categorized into signals, background, etc, and integrated, and each data is sent to the specified port.  
DataViewerBase accesses and acquires data from those ports, and integrates them and displays them on the GUI.