# DataViewerBase

**DataViewerBase** パッケージはリアルタイムで測定された画像を確認するためのGUIやツールを提供します。  
SPring-8内のSACLAにてカメラで測定されたデータを積算し、それらを確認するために作成されました。

## システム要件

DataViewerBase を使用するには以下のパッケージおよびバージョンが必要です。  
これらのうち olpy および stpy はSACLAのHPC上で利用されるモジュールです。  
動作確認をするだけであれば必要ありません。

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

## 使用方法

1. ターミナル（コマンドプロンプト）を２つ開きます。便宜上、これらを A, B と呼ぶことにします。
2. ターミナルAでDataViewerBaseディレクトリ（フォルダ）に移動します。
   * これによってデータベースからのデータの取得が開始されます。
3. python getDatawithOLPY.pyを実行します。
4. ターミナルBでDataViewerBaseディレクトリ（フォルダ）に移動します。
5. python main.pyを実行します。
   * これによってGUIが起動します。

### DataViewerBaseの機能

* Startボタンを押すことでデータの積算・表示が開始されます。
* Clearボタンを押すことでデータをクリアします。
* Saveボタンを押すことでその時点でのデータを保存します。
* Windowボタンを押すことで、別のウィンドウでデータを確認できます。
  + 2017/10/15 時点では特に必要のない機能です。

## ポートなどの設定

ポートなどの設定は二つのファイルで行います。

### /config\_getdata.json

getDatawithOLPY.pyで利用する設定です。

# /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

DataViewerBase.pyで利用する設定です。

# /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

## 構成

DataViewerBaseパッケージの構成は次図の通りです。  
SACLAのデータベースからデータを取得する機能はgetDatawithOLPY.pyスクリプトによって行われます。  
データの表示はDataViewerBaseクラス（GUI）によって行われます。

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.

## データフロー

SACLAのサーバに蓄積された測定データをgetDatawithOLPY.pyによって取得します。  
それらをシグナルとバックグラウンド、などに分類して積算し、指定されたポートにデータを送ります。  
それらのポートに DataViewerBase がアクセスしてデータを取得し、さらに DataViewerBase の内部で積算したものをGUI上に表示します。