



# **Advanced Virgo dataDisplay**

## **Users Manual**

D. Verkindt

VIR-0047C-21

<https://tds.ego-gw.it/ql/?c=16295>

3 June 2022

## Change Record

Version	Date	Section Affected	Reason / Remarks
v1	19/11/2019	All	First version
v1	03/12/2019	Appendix A	Added text of video tutorials
v1	24/04/2020	All	Additional features+Appendix
v1	23/06/2020	All	Added ref plots and options
v1	20/11/2020	All	Added options + final modifs
v2	03/05/2021	All	Update pictures and text. Added Useful tools section
v3	03/06/2022	All	Add new options and features for channels and plots

# Table of Contents

1	Introduction.....	5
2	Input parameters and Configuration.....	6
3	Main panel.....	7
4	Inputs : sources of data .....	9
5	Main browser: Channels and Plots.....	10
5.1	Channels management.....	11
5.2	Plots generation .....	12
5.3	Plots management .....	13
5.4	TIME plot .....	14
5.5	BRMSTIME plot.....	15
5.6	FFT plot .....	16
5.7	FFTTIME plot.....	17
5.8	TRFCT plot.....	18
5.9	TRFCTTIME plot .....	19
5.10	COHE plot .....	20
5.11	COHETIME plot .....	21
5.12	1D-DISTRIB plot .....	22
5.13	1DTIME plot.....	23
5.14	2D-DISTRIB plot .....	24
5.15	RAW-IMAGE plot .....	25
5.16	RAWTIME plot .....	26
5.17	AUDIO plot.....	27
6	Reference Plots.....	28
7	Tools and Options.....	29
7.1	The Options panel .....	29
7.2	The Sliders panel .....	31
7.3	The debug panel.....	32
8	Plots Root menus.....	33
8.1	The plot's line menu .....	33
8.2	The X axis and Y axis menu .....	34
9	Useful Additional Tools .....	35
9.1	rawframe.C.....	35
9.2	rawview.C .....	35
9.3	sound2frame .....	36

9.4	export2ascii.C .....	37
9.5	asciiview.C .....	37
10	Bibliography.....	38
11	Appendix A : Video tutorials.....	38
11.1	Video 1: Starting dataDisplay .....	38
11.2	Video 2: Inputs.....	38
11.3	Video 3: Mainbrowser : Channels management.....	39
11.4	Video 4: Mainbrowser: Combine channels .....	39
11.5	Video 5: Mainbrowser: Plots management.....	39
11.6	Video 6: Edit Plot TIME.....	40
11.7	Video 7: Edit Plot FFT.....	41
11.8	Video 8: Edit Plot TRFCT and COHE .....	41
11.9	Video 9: Edit Plot FFTIME, COHETIME, 1DTIME .....	42
11.10	Virgo 10: Start/Stop/Pause and mainpanel infos.....	42
11.11	Video 11: Ref Plots management .....	42
11.12	Video 12 : Options .....	43
11.13	Video 13 : Help and help ballons.....	43
12	Appendix B : dataDisplay Configuration.....	44
13	Appendix C : Associated tools .....	47
14	Appendix D : Needed libraries.....	47

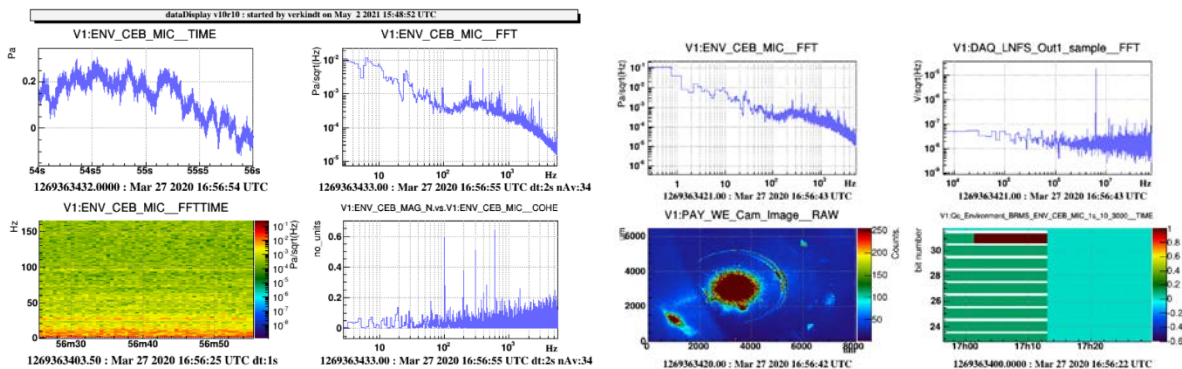
# 1 Introduction

This document describes the version v10r12 of the dataDisplay tool that is devoted to the access, management and display of the frame formatted data of gravitational waves detectors like Virgo or LIGO.

The dataDisplay can read those frame formatted data online or offline and can display various types of plots that help in monitoring the interferometer, investigating noise features or checking the various interferometer's signals or environment's signals provided by the data acquisition system.

A Graphical User Interface, based on the Xforms library, provides the main panels that allow to manage the data access, the channels selection and the plots settings.

Plots generation and manipulation, based on the ROOT library, have specific features that allow to do various operations on several plots at the same time, to superpose additional plots or to extract any plot's data down to an ascii file.



Example of dataDisplay plots

- Section 2 describes the configuration file dy.cfg, where are saved all the parameters defining the data access, the channels selection and the plots features.
- Section 3 describes the main panel features.
- Section 4 is devoted to the main browsers for channels and plots management.
- Section 5 describes the Edition panels used to set the plots parameters.
- Section 6 describes how to superpose reference plots on the plots currently displayed.
- Section 7 describes the various options, debugging features and checks that can be set from the main panel "Tools" button.
- Section 8 describes the features and menus available through a right-click on the plots or on their axis.
- Section 9 is a documentation about additional useful tools (like the ROOT script rawview.C) provided within the dataDisplay package.
  
- Appendix A contains the texts for the video training of the dataDisplay.
- Appendix B contains the description of the keywords used in a dataDisplay configuration file.
- Appendix C is a short description of some additional tools that can be associated to dataDisplay.
- Appendix D is the list of libraries used by the dataDisplay package and some information about where to download the dataDisplay package and the needed libraries.

## 2 Input parameters and Configuration

Each time you push on the Start button of the main panel of the dataDisplay, a configuration file **dy.cfg** is created, that contains all the parameters defining the data access, the channels selection and the plots chosen.

To start a dataDisplay directly with this configuration, you can then just type “**dataDisplay dy.cfg**”. All the keywords of a typical configuration dy.cfg are described in Appendix B.

In addition to the name of a configuration file, you can provide various input command line parameters. You can get the description of those parameters by typing “**dataDisplay -h**” and you will get the following description:

**Usage: dataDisplay [config] -f [start GPS] -l [duration] -o [filename] -t [title] -nocm -b -q -h**

[config]	(default=NULL)	= name of configuration file
-f [start GPS]	(default=now)	= start GPS time
-l [duration]	(default=infinite)	= number of seconds of data to read
-b		= dataDisplay run in background
-q		= quit when dataDisplay stops
-o [filename]	(default=config)	= output picture (.jpg, .png, .gif, .eps, or .root files)
-t [title]	(default=user/date)	= title chosen for the plots canvas
-name [dynamy]	(default=dy)	= name of the dataDisplay (used in the ROOT canvas and Cm name)
-noplot		= plots not displayed but a dd.root file produced
-nocm		= no use of Cm connection
-debug [level]	(default=0)	= debug level

Examples of dataDisplay command line:

**dataDisplay dy.cfg -f 1240002011 -l 20 -b -q -o myplots.png**

This will take the configuration dy.cfg but will use a new start time 1240002011 and a new duration of 20 seconds. It will run in background mode (no plot shown) and will quit once all data have been read. The produced plots are in a ROOT canvas saved as png picture.

**dataDisplay dy.cfg -b -q -noplot**

This will take the configuration dy.cfg, run in background mode (no plot shown) and will quit as soon as all data have been read. No plot is visible but a file dy.root is output and contains the plots.

**dataDisplay -nocm**

The dataDisplay is started without using the Cm library. Then, online connection to a frames provider is not possible, only input data from files or shared memory is possible. This is an old option, that allows using the dataDisplay even if it was compiled without Cm.

**datadisplay myconfig.cfg -t “plots of myconfig”**

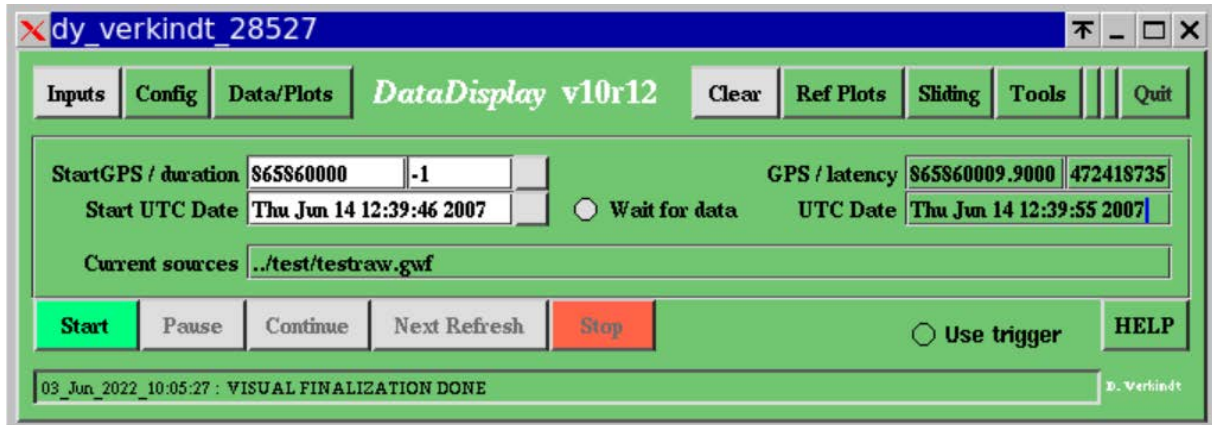
This will take the config myconfig.cfg but replacing the default upper title of the ROOT Canvas by the string “plots of myconfig”.

**dataDisplay -name “mydatadisplay”**

The dataDisplay is started with a new Cm name “mydatadisplay” which will be also the upper title of the ROOT Canvas.

### 3 Main panel

The main panel is the GUI panel that appears when you start the dataDisplay. It contains several buttons, menu buttons or input fields that you can use to access to the various features of the dataDisplay or to configure the way input data will be read.



**Inputs:** this menu button gives access to the list of possible sources from which the dataDisplay can read input data. It is described in more details in section 4.

**Config:** this menu button allows to load or save a dataDisplay's configuration file.

**Data/Plots:** this button opens the main browser panel that allows to select the input data channels and to edit the plots you made with them. This panel is described in more details in section 5.

**Clear:** this button deletes the ROOT canvas that contains the plots. This allows you to reset the various zooms or changes you made on the plots interactively.

**Ref Plots:** this menu button allows you to set, load, save or edit the current plots as reference plots that will be superposed persistently on the plots, the next time you click on Start. The reference plots are fully described in section 6.

**Tools:** this menu buttons gives access to various options or checks. The items of this menu are described in section 7.

**Quit:** this button stops the dataDisplay, after confirmation.

**Start GPS:** the GPS time of the first input data to read

**Duration:** number of seconds of input data to read

**Start UTC Date:** the UTC date and time of the first input data to read. You can provide any format among the following ones: Wed Sep 18 14:19:41 2019 , Sep 18 14:19:41 2019 , Sep 18 2019 14:19:41 , 2019/09/18 14:19:41 , 2019-09-18 14:19:41

**GPS:** this field shows the GPS time of the current data read

**Latency:** number of seconds between the current GPS time and the GPS time of the data read

**UTC Date:** UTC date and time of the data read.

**Wait for data:** if this button is ON, the dataDisplay waits for next data once it has reached the end of the input data file. This allows to read a file which is regularly updated.

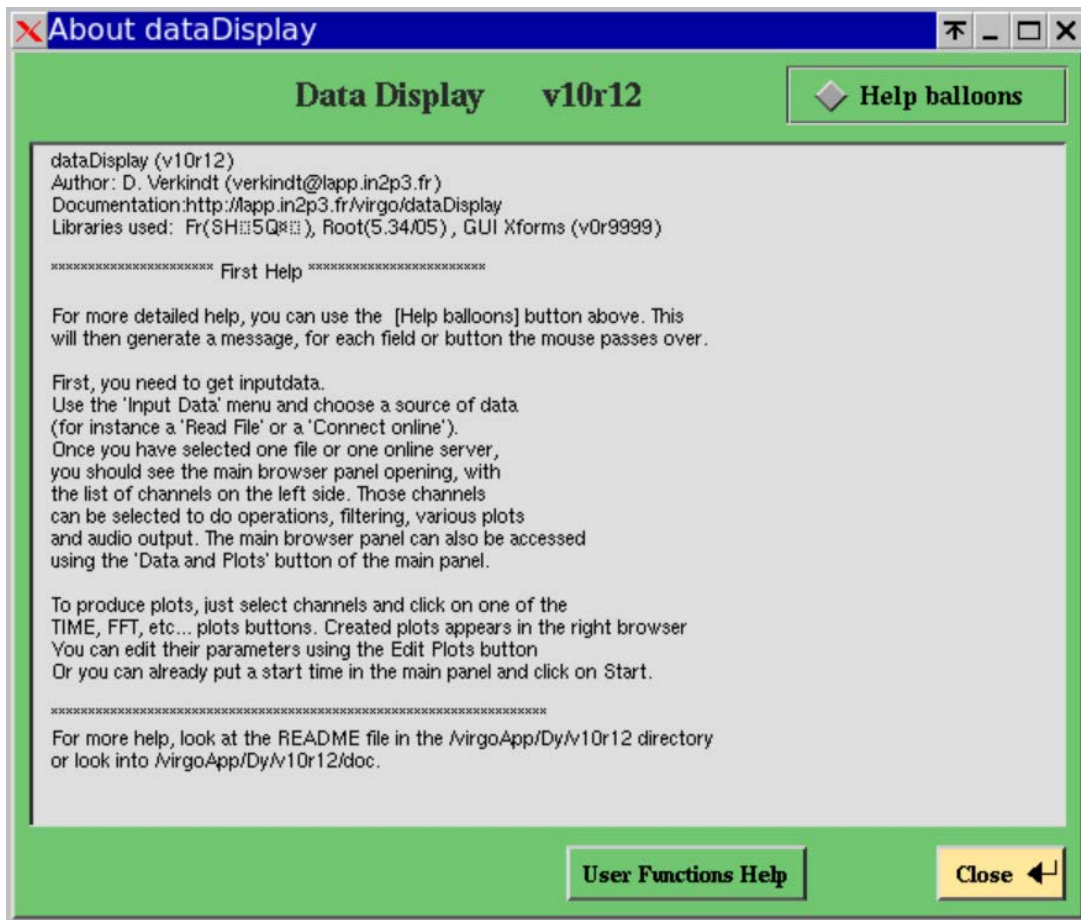
**Current sources:** this field shows the list of the input sources from which input data are currently read. If you directly change the first input of this list, the channels and plots using this input will automatically be updated.

**Start / Pause / Continue / Next Refresh / Stop :** buttons to start, pause, continue and stop the reading of the input data. The “Next Refresh” button is like “Continue” but with an automatic Pause once a plot has been updated with new data.

**Use trigger:** if this button is ON, the trigger condition defined in the mainbrowser panel is used to select the data periods used by the plots.

**HELP:** this button opens a panel (see below) providing a first help about how to use the dataDisplay.

**Info Line:** Show the last information available about the dataDisplay actions. Clicking on this line opens a log containing all the log messages that appeared in this “Info line” since the start of the dataDisplay.



The Help panel of the dataDisplay



## 4 Inputs : sources of data

The “Inputs” menu button of the main panel shows a menu with various possible sources of data. Among them, the most commonly used are the “Read FFL”, the “Read Files” and the “Connect Online” but you can try the other items in special cases like remote access to data stored on disk. Below is the full list of items of the “Inputs” menu:

**Read FFL:** this item opens a file browser panel showing by default the list of ffl files (\*.ffl) available in /virgoData/ffl directory. ffl files are lists of frame files, specific to Virgo and LIGO frame format, which allow to access to a large set of files as if accessing to a single file.

**Read Files:** this item opens the file browser panel showing by default the list of frame files (\*.gwf) available in the local directory. You can change the directory either by using a “single click” navigation within the files list or by typing the directory path in the directory input field. Then, select all the files you want to read and click on OK. You can also type directly the path/name of the file in the filename input field and click on OK.

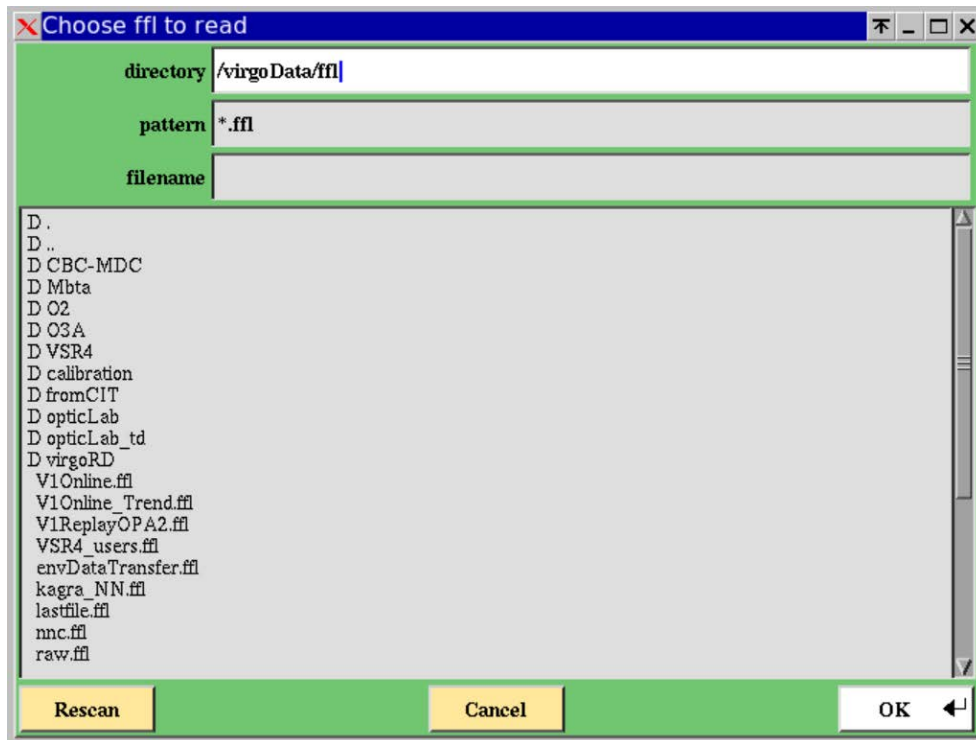
**Connect Online:** this item opens a panel showing the data providers available in the Cm Domain chosen (default is “Cascina”). Once you have selected one data provider, click on OK. This action sends a request to the data provider that sends back a list of channels available online. Then, clicking on the Start button, a channel request is sent and the data provider will then send only the channels needed for the plots. This type of data retrieval use Cm, which is a Virgo specific library allowing to exchange data easily between two servers of the DAQ or between a dataDisplay and a data provider connected at any point of the data collection architecture.

**Connect to dataSender:** this item opens a panel allowing to choose one of the dataSenders available in the current Cm domain. A dataSender is similar to the data providers previously described but it sends data read from files instead of sending data online. Each dataSender receives a request for a full channels list as soon as you first connect to it. Then, it can be used by the dataDisplay to get data read from any files located on the remote disks.

**Read Shared Memory:** this item opens a panel allowing the dataDisplay to connect to a shared memory available on the same computer as the one where the dataDisplay was started. If this shared memory contains frame formatted data, it can be read by the dataDisplay.

**Read Audio Wave File:** this item opens the file browser. If you select a .wav audio file, its data will be frame formatted and a channel will be created. You can then select it to do various plots.

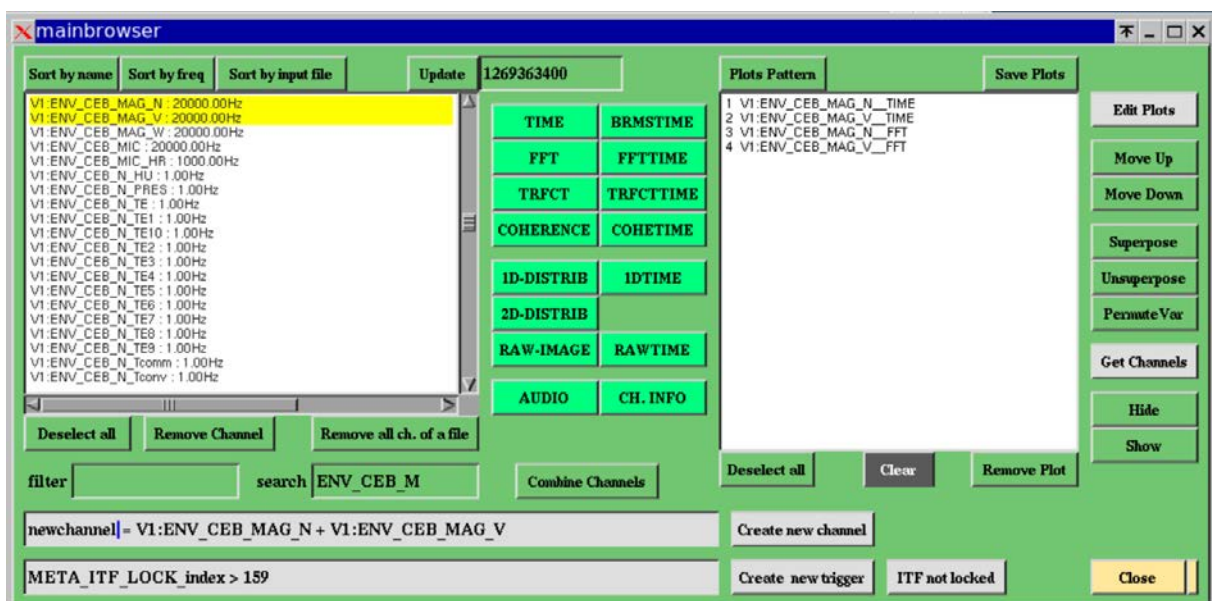
**Read Ascii File:** this item opens the file browser. If you select a .txt file in ASCII or CSV format, its data will be frame formatted and a channel will be created. You can then select it to do various plots.



The input file browser panel

## 5 Main browser: Channels and Plots

This panel opens when you click on the “Channels/Plots” button of the main panel. It is the central node of the dataDisplay where you can select channels in the channels browser, choose the type of plot you want for those channels and edit the plots you created. The various browsers, buttons and input fields of this panel are described in the subsections below.



The mainbrowser panel

## 5.1 Channels management

**Channel browser:** once you have selected one or several inputs, the channel browser shows all the data channels available. Each line shows the name of the channel, the sampling frequency of the channel and the input source of the channel.

**Sort by name, Sort by frequency, Sort by input file:** allow to sort the channels of the channels browser, according to the name of the channel or to the sampling frequency of the channel or to the source of the channel.

**Update:** this button updates the channel list by looking at the channels available at the GPS time provided in the input field near to the button.

**Deselect all:** this button deselect all the channels in the list.

**Remove channel:** this button removes from the list any selected channel that was created by the user.

**Remove all ch. of a file:** Remove from the channels list all the channels, which have the same source as the selected channels.

**Filter:** this field allows you to apply a filter in the list of channels and to put at the top of the list the filtered channels as soon as you type "Enter" key on your keyboard. Examples: If you put "ENV\_\*", all the channels whose name starts with ENV\_ will be put at the top of the list. If you put "\*MC\*temp", all the channels whose name contains "MC" and finishes with "temp" will be put at the top of the list.

**Search:** it allows finding in the channels list the first channel that fulfil the request. Example: typing ENV will show you the channel ENV\_BS\_ACC\_Z. If you continue to type, for instance ENV\_CE, you will get ENV\_CEB\_DUST\_OP3UM that is the first channel whose name starts with ENV\_CE.

**Combine Channels:** this button takes all the selected channels and show a sum of them in the "Create new channel" field. You can then modify this operation using the usual +,-,x,/ operators and parenthesis but also using any mathematical function compliant with standard C math library. You can also remove some of the channels or add channel names you know.

**Create new channel:** This button takes the formula you have put for the combination of channels and create a new channel. If your formula starts with "mychannel =", the new channel will appear in the channels list as "zz\_mychannel". You can then use it to do a plot or remove it from the channels list with the button "Remove Channel".

Then, each time you select such a created channel, you will see appearing in the "Channel Combination" field the formula that defines it. Be aware that, as soon as you change the formula, this change it taken into account even if you have not used again the button "Create a new channel" and is applied to the plots already made with the newly created channel. See video tutorial for more explanation.

**Create New Trigger:** this button creates a new trigger based on the formula provided in the corresponding input field. Example: if you put in this field "ENV\_CEB\_MIC < 0.05 and ENV\_CEB\_MAG\_N > 0.01" (requesting acoustic channel to be below a threshold of 0.05 and magnetic channel to be above 0.01), you will then see yellow boxes in the TIME plots for any time period where the formula is not verified. In addition, you will see FFT plots computed only over the time periods where the formula is verified. The "ITF Not locked" button puts back the default trigger which is META\_ITF\_LOCK\_index > 159

## 5.2 Plots generation

**TIME:** clicking on this button creates a TIME plot for each channel selected in the channels list. Such plot shows versus UTC time the data samples of the channel.

**BRMSTIME:** clicking on this button creates a BRMSTIME plot for each channel selected in the channels list. Such plot shows versus UTC time the rms of the channel computed in a given frequency band. This computation uses the FFT of the signal and depends on the frequency resolution of this FFT.

**FFT:** clicking on this button creates a FFT plot for each channel selected in the channels list. Such plot shows the module of the Fourier Transform of the channel (amplitude spectral density of the noise in this channel).

**FFTTIME:** clicking on this button creates a FFTTIME plot for each channel selected in the channels list. Such plot shows versus UTC time the module of the Fourier Transform of the channel. This is time-frequency plot or a spectrogram, very helpful to view in a single eyes-shot the time and frequency characteristics of a signal. It has some constraints that are described in more details in section about edition of plot parameters.

**COHE:** clicking on this button creates a COHE plot for each couple of channels selected in the channels list. Such plot shows versus frequency the coherence between two channels. The coherence is defined as the cross-correlation of the two channels divided by the square of the auto-correlation of each channel.

If the channels have different sampling frequencies, the channel with the highest sampling frequency is downsampled to reach the lowest sampling frequency.

**COHETIME:** clicking on this button creates a COHETIME plot for each couple of channels selected in the channels list. Such plot shows versus UTC time the time evolution of the coherence between two channels.

**TRFCT:** clicking on this button creates a TRFCT plot for each couple of channels selected in the channels list. Such plot shows versus frequency the module and phase of the transfer function between two channels. The transfer function is defined as ratio between the FFT of each channel.

If the channels have different sampling frequencies, the channel with the highest sampling frequency is downsampled to reach the lowest sampling frequency.

**TRFCTTIME:** clicking on this button creates a TRFCTTIME plot for each couple of channels selected in the channels list. Such plot shows versus UTC time the time evolution of the module of the transfer function between two channels.

**1D-DISTRIB:** clicking on this button creates a 1D plot for each channel selected in the channels list. Such plot shows the distribution of the sample values of the channel. By default, the plot shows also basic statistics like the mean and the rms values.

**1DTIME:** clicking on this button creates a 1DTIME plot for each channel selected in the channels list. Such plot shows versus UTC time the time evolution of the distribution of the sample values of the channel.

**2D-DISTRIB:** clicking on this button creates a 2D plot for each couple of channels selected in the channels list. Such plot shows the sample values of one channel versus the sample values of the other channel.

If the channels have different sampling frequencies, the channel with the lowest sampling frequency is oversampled to reach the highest sampling frequency.

**AUDIO:** clicking on this button creates a fictive plot that is in fact a time segment of the channel's data that will be recorded in an audio file or that you can hear online as soon as you have clicked on the "Start" button of the main panel.

**CH. INFO:** this button provides a short description of each channel selected. This description is available in a dedicated panel, in the log file (Info line at the bottom of the main panel) and in the xterm window where the dataDisplay was started. This information is taken either from the VIC channels database or from a dedicated file `/virgoData/Dy/chlist.txt`

## 5.3 Plots management

Each time you use one of the buttons above, you create one plot for each channel selected. Those plots are listed in the plots browser. Clicking on several plots selects them. You can then use the buttons on the right of the plots browser to do some action on the selected plots:

**Plots pattern:** define the configuration of the ROOT Canvas that shows the plots. Clicking on it opens an input field. If you type 2x3 in this input field, the plots will appear in a set of pads of 2 columns by 3 rows, except if there are more than 6 plots to display. In this case, the automatic pattern is applied.

**Save Plots:** this button saves the ROOT canvas and all its plots in a file (default is `dy.root`) which can be a ROOT file or a jpg or png or gif file.

**Deselect all:** this button deselects all the plots selected

**Clear:** this button removes all the plots from the browser. Take care, no undo is possible and no confirmation is asked.

**Remove Plot:** this button removes from the browser all the plots selected.

**Edit Plots:** this button opens an edition panel associated to the type (TIME, FFT, 1D, etc...) of the last plot selected. All the changes done in this edit panel applies to all the selected plots of the same type as the last plot selected.

**Move Up / Move Down :** allows to move up or down the selected plots in the plots list

**Superpose / Unsuperpose :** this button marks as superposed all the selected plots of the same type. Superposed plots will then appear in the same ROOT canvas pad. For instance, if you have selected two TIME plots and two FFT plots, whatever their position in the plots list, it will superpose the two TIME plots in a pad and the two FFT plots in another pad.

**PermuteVar:** this button allows to permute the channels in a TRFCT, COHE or 2D plot. This is useful mainly because when a TRFCT plot is created it takes the selected channels in the order provided by the channels browser, an order not fully under user's control.

**GetChannels:** clicking on this button gets all the channels used in the selected plots and make them selected in the channels browser. This is useful especially when you have a set of plots of a given type

(for instance TIME) and you want, for the same channels, a set of plots of another type (for instance FFT).

**Hide / Show** : this button puts all the selected plots in a Hide state (or Show state). A plot in a Hide state is kept in the plots list but is not computed and not displayed when you click on the “Start” button of the main panel.

Each time you click on the “Edit Plot” button, an edition panel is opened, specific to each plot’s type. Below is a description of each of those panels.

## 5.4 TIME plot

The edit panel of the TIME plot has the following input fields and buttons:

**Time window**: number of seconds of data displayed

**Sampl. Freq.** : sampling (or resampling) frequency of the data displayed

**Minmax**: if this flag is ON, what is displayed is only the min and max of the data resampled

**Time Shift**: percentage of time window. It is the size of data stretch read between each plot’s update

**ymin / ymax**: the min and max values of the data shown on the plot or the Y axis span.

**y offset**: the offset to add to the data before displaying them in the plot

**y scaling factor**: scale factor to apply before displaying the data in the plot. It applies before the offset.

**Band-pass filter (fmin/fmax)**: frequency min and max of the band-pass filter applied to the data

**logx**: put log scale on the x axis of the plot

**logy**: put log scale in the y axis of the plot

**gridx**: show vertical grid lines along the x axis

**gridy**: show horizontal grid lines along the y axis

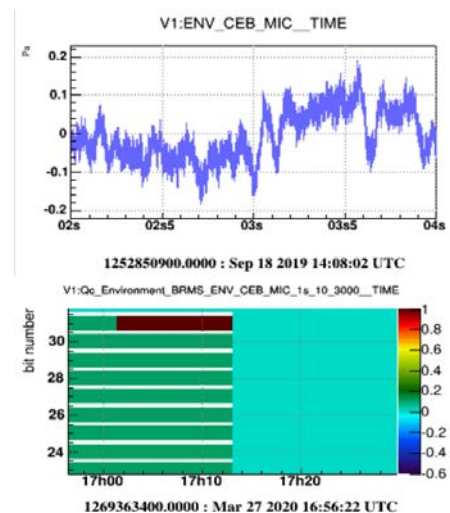
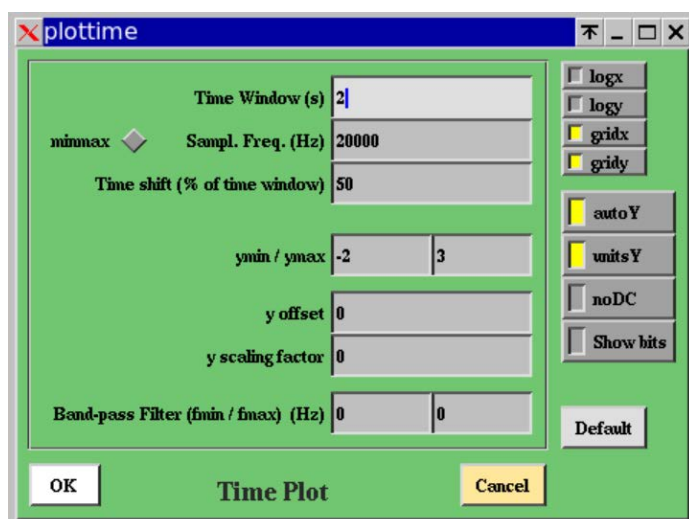
**autoY**: sets automatically the min and max on Y axis according to the current data displayed

**unitsY**: show on y axis calibrated data instead of raw ADC counts

**noDC**: subtract from the data the mean value computed over the data samples currently displayed

**Show bits**: show, for each data sample the value of its first 32 bits instead of the sample value itself  
(blue = no data, green = bit value is 0, red = bit value is 1)

**Default**: sets default values for all the fields of the edition panel



## 5.5 BRMSTIME plot

The edit panel of the BRMSTIME plot has the following input fields and buttons:

**Time window:** number of seconds over which band rms values are computed and displayed

**BRMS time window:** number of seconds of data used to compute each FFT used to compute the BRMS.

This determines the frequency resolution of the FFTs used.

**Frequency resolution:** width (in Hz) of each bin of the spectrum used to compute the band rms. This is the inverse of the size of the FFT time window.

**Time resolution:** width (in sec) of each bin on the X (time) axis. If the time resolution is 10 sec and if each FFT is computed over 1 sec (frequency resolution of 1 Hz), with an overlap of FFTs of 50%, then each bin on X axis will contain the average (or the median) over 20 FFTs. The time resolution is also the BRMS time serie sample rate.

**Sampl. Freq. :** sampling or resampling frequency of the input channel

**Time Shift (% of FFT time window):** shift (in sec) on input data between two consecutive FFTs

**Refresh Period (% of time window):** size of the data stretch read between two plot's updates

**Freq band min/max:** min and max of the frequency band over which the BRMS is computed

**ymin / ymax :** min and max BRMS values shown on the plot

**logx:** put log scale on the x axis of the plot

**logy:** put log scale in the y axis of the plot

**gridx:** show vertical grid lines along the x axis

**gridy:** show horizontal grid lines along the y axis

**autoY:** sets automatically the min and max on Y axis according to the current data displayed

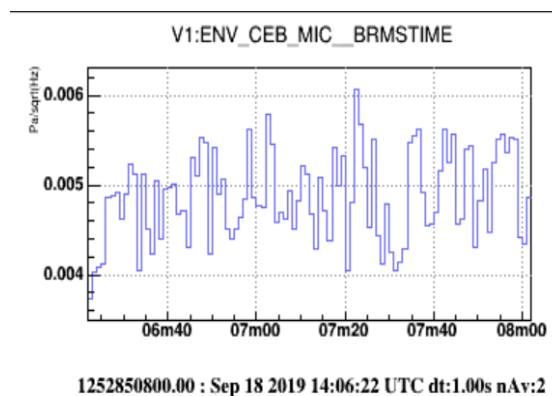
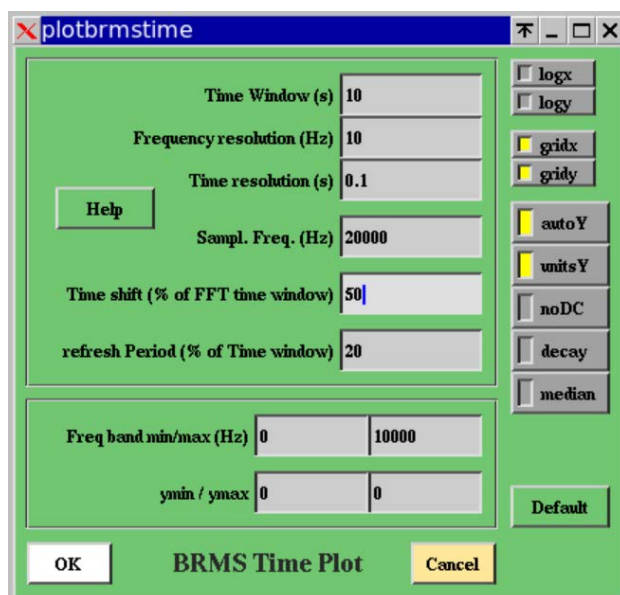
**unitsY:** show on y axis the BRMS computed with calibrated data instead of raw ADC counts

**noDC:** subtract from the input data the mean value computed over each FFT data samples

**decay :** use a moving average when computing the averaged FFTs

**median :** use a median FFT instead of an averaged FFT to compute the BRMS values

**Default:** put default values for all the parameters of the edit panel



## 5.6 FFT plot

The edit panel of the FFT plot has the following input fields and buttons:

**Time window:** number of seconds of data over which the Fast Fourier Transform is computed. Each data stretch in this time window is first multiplied by a tapering Hanning window.

**Sampl. Freq. :** sampling (or resampling) frequency of the data passed to the FFT. An anti-aliasing filter (low-pass Butterworth of order 16) is applied in case of resampling

**Time Shift (% of time window):** the overlap between two successive FFT time windows (usually 50%)

**Number of FFTs to average:** number of modules of the FFT (amplitude spectral density) to average

**refreshPeriod (in number of FFTs):** number of FFT averaged between each plot refresh

**freqmin / freqmax:** min and max frequency of the FFT frequency band displayed

**ymin / ymax:** the min and max of the FFT amplitude displayed or the Y axis span

**band-pass filter (fmin/fmax):** Frequency min and max of the pass-band filter (a double Butterworth filter of order 16) applied to the data before computing the FFT.

**logx:** put log scale on the x axis of the plot

**logy:** put log scale in the y axis of the plot

**gridx:** show vertical grid lines along the x axis

**gridy:** show horizontal grid lines along the y axis

**autoY:** sets automatically the min and max on Y axis according to the current data displayed

**unitsY:** show on y axis calibrated data instead of raw ADC counts

**noDC:** subtract from the data the mean value computed over the data samples of the last FFT

**dB:** sets the Y axis in decibels units

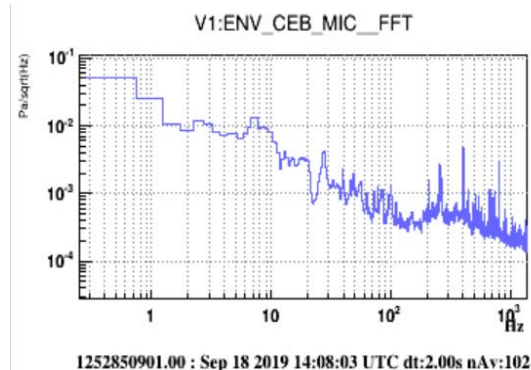
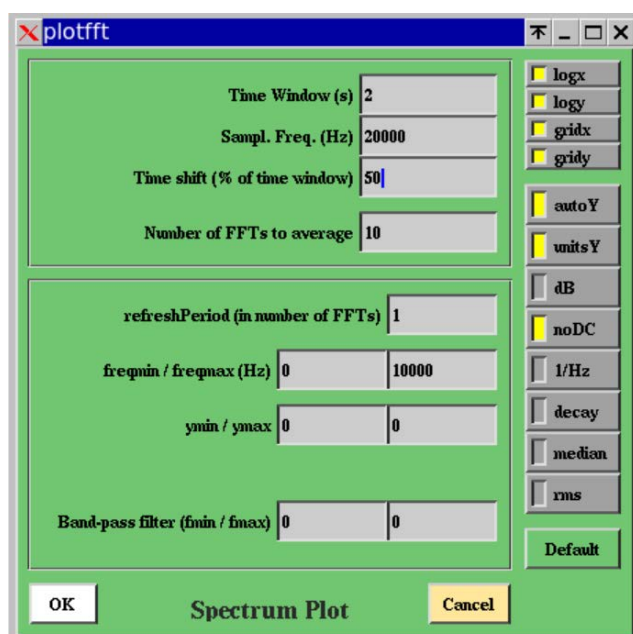
**1/Hz:** show the FFT module in 1/Hz units instead of 1/sqrt(Hz) units

**decay:** the FFT displayed is a moving averaged FFT, with a depth set by the “Number of FFT to average”

**median:** Compute a median FFT module instead of an averaged FFT module. Median is computed over N FFTs, where N is the “Number of FFT to average”. If decay is ON, median becomes a moving median.

**rms:** The integrated rms computed over the FFT module is also displayed. It is computed from the Nyquist frequency down to 0 Hz. Value of this curve at 0 Hz is the rms of the signal.

**Default:** sets default values for all the fields of the edition panel





## 5.7 FFTTIME plot

The edit panel of the FFTTIME plot has the following input fields and buttons:

**Time window for Time-Freq plot:** number of seconds over which the time-frequency plot is done

**Frequency resolution:** width (in Hz) of each bin on the Y (frequency) axis

**Time resolution:** width (in sec) of each bin on the X (time) axis. If the time resolution is 10 sec and if each FFT is computed over 1 sec (frequency resolution of 1 Hz), with a FFT overlap of 50%, then each bin on X axis will contain the average (or the median) over 20 FFTs.

**Sampl. Freq. :** sampling or resampling frequency of the input signal of the plot

**Time Shift (% of FFT time window):** time shift in input data between two consecutive FFT computation. It is a percentage of the FFT time window. The size of the FFT time window is the inverse of the frequency resolution. If the time shift is 30%, it means that the overlap between two consecutive FFT time windows is 70%.

**Refresh Period (% of time window):** quantity of data read between two consecutive plot's update.

**Zoom freq min/max:** min and max of the frequency band shown on the Y axis (this does not prevent to make an unzoom once the plot is done and to see all the frequency band between 0 and the Nyquist frequency).

**zmin / zmax (color scale):** min and max values of the color scale which indicates the FFT module in the time-frequency plot. You can improve the contrast by reducing the span between zmin and zmax. If  $zmin \geq zmax$ , then the option flag autoZ is automatically activated.

**Help:** this button opens a help panel that explains how is computed the time-frequency plot

**logx:** put log scale on the x axis of the plot

**logy:** put log scale in the y axis of the plot

**logz:** put log scale on the z axis of the plot

**gridx:** show vertical grid lines along the x axis

**gridy:** show horizontal grid lines along the y axis

**gridz:** show grid lines along the z axis when the plot is done with LEGO or SURF instead of COL option.

**autoZ:** sets automatically the min and max on z axis according to the current data displayed

**unitsZ:** show on the z axis calibrated data instead of raw ADC counts

**noDC:** for each computed FFT, subtract first from the data the mean value computed over the data samples used for this FFT

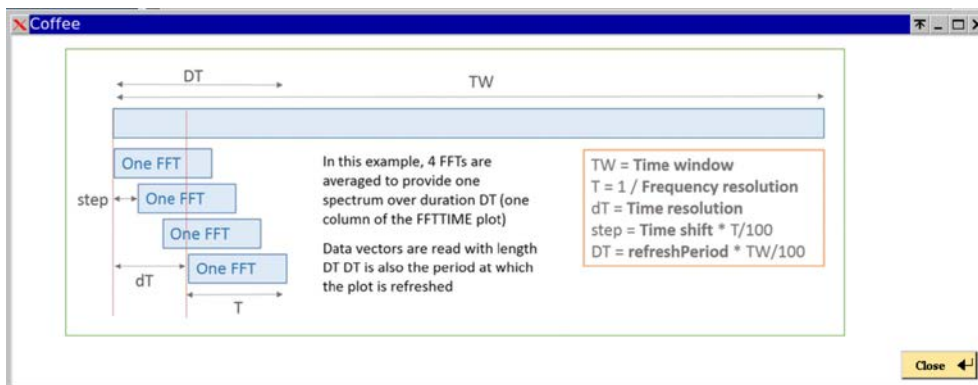
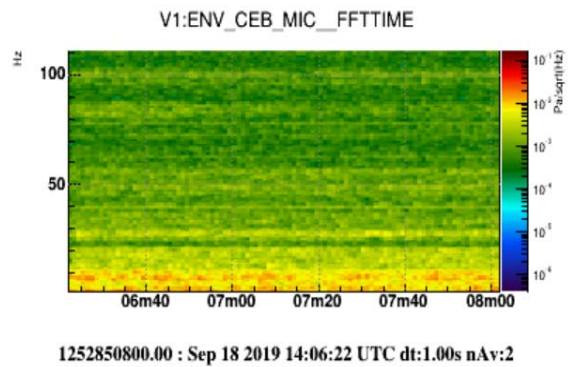
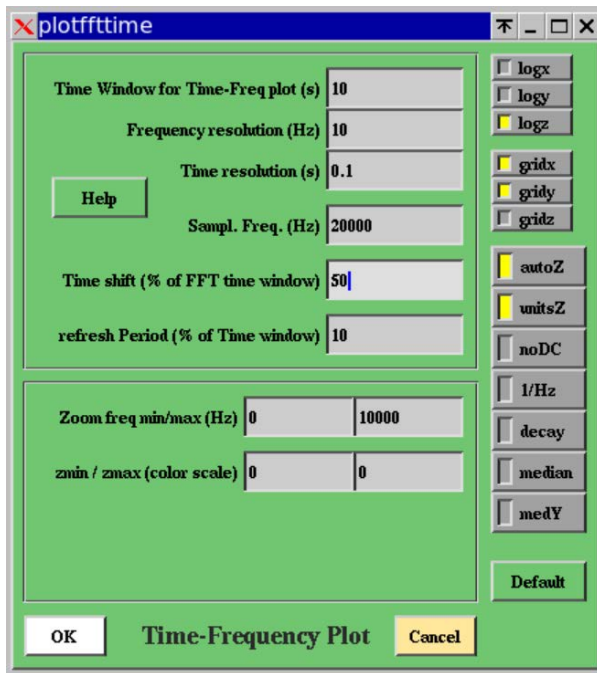
**1/Hz:** Put as FFT module the usual units instead of the  $1/\sqrt{\text{Hz}}$  units of a noise amplitude density.

**decay:** use a moving average when computing the averaged FFTs.

**median:** if this flag is ON, a median FFT is computed instead of an averaged FFT

**medY:** normalize each value at coordinated X,Y of the plot by the median value computed over all the values along a given Y coordinate. This allows improving the visibility of any time variation of the signal FFT, like glitches.

**Default:** put the default values for each editable parameter.



Help panel of the FFTTIME plot

## 5.8 TRFCT plot

The edit panel of the TRFCT plot has the following input fields and buttons:

**Time window:** number of sec of data used to compute each transfer function

**Channel 1 Sampl Freq :** sampling or resampling frequency of the numerator channel

**Channel 2 Sampl Freq:** sampling or resampling frequency of the denominator channel

**Time shift (% of time window):** shift on the data read between each transfer function computation

**Number of FFTs to average:** number of transfer function computation to average

**Refresh Period:** number of transfer function computation between each plot's update

**freqmin / freqmax:** min and max of the frequency band displayed in the plot

**yMin / yMax:** min and max value of the transfer function module displayed

**Band-pass filter (fmin/fmax):** min and max frequencies of the pass-band filter applied on the two input channels before each transfer function computation

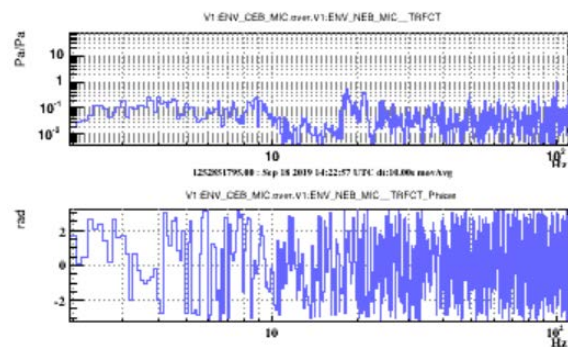
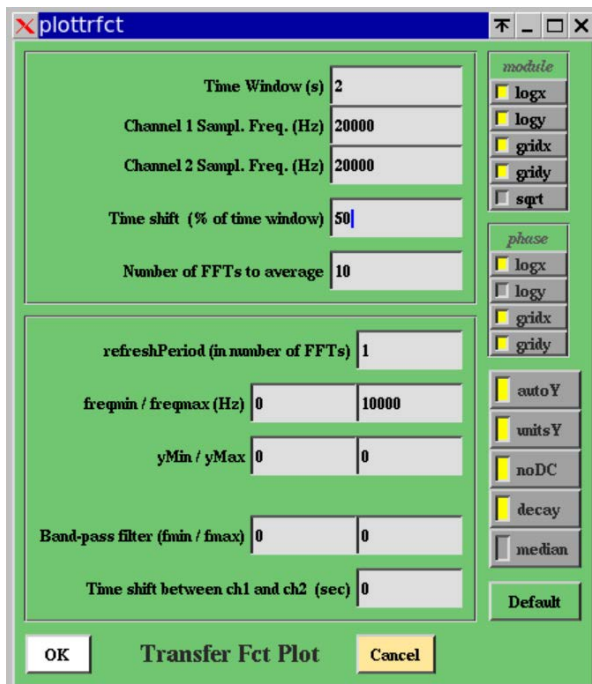
**Time shift between ch1 and ch2:** time shift (in seconds) applied between the two input channels before each transfer function computation

**Logx:** put log scale on the x axis of the plot of the transfer function module

**logy**: put log scale in the y axis of the plot of the transfer function module  
**gridx**: show vertical grid lines along the x axis of the plot of the transfer function module  
**gridy**: show horizontal grid lines along the y axis of the plot of the transfer function module  
**sqrt**: use correlation instead of the square of the correlation in the definition of the transfer function

**logx**: put log scale on the x axis of the plot of the transfer function phase  
**logy**: put log scale in the y axis of the plot of the transfer function phase  
**gridx**: show vertical grid lines along the x axis of the plot of the transfer function phase  
**gridy**: show horizontal grid lines along the y axis of the plot of the transfer function phase

**autoY**: sets automatically the min and max on Y axis according to the current data displayed  
**unitsY**: show on y axis calibrated data instead of raw ADC counts  
**noDC**: subtract from each input channel its mean value computed over the data samples used to compute the transfer function.  
**decay**: use a running average (or median) when averaging the transfer functions  
**median**: compute a median transfer function instead of an averaged transfer function  
**Default**: put the default values for each parameter of the plot's edit panel



## 5.9 TRFCTTIME plot

The edit panel of the TRFCTTIME plot has the following input fields and buttons:

**Time window for Time-Freq plot**: number of seconds over which the time-frequency plot is shown

**Frequency resolution**: width (in Hz) of each bin on the Y (frequency) axis

**Time resolution**: width (in sec) of each bin on the X (time) axis. If the time resolution is 10 sec and if each FFT is computed over 1 sec (frequency resolution of 1 Hz), the each bin on X axis will contain the average (or the median) over 10 FFTs.

**Channel 1 Sampl Freq**: sampling or resampling frequency of the numerator channel

**Channel 2 Sampl Freq**: sampling or resampling frequency of the denominator channel

**Time Shift (% of FFT time window):** time shift in input data between two consecutive FFT computation. It is a percentage of the FFT time window. The size of the FFT time window is the inverse of the frequency resolution. If the time shift is 30%, it means that the overlap between two consecutive FFT time windows is 70%.

**refreshPeriod (% of time window):** quantity of data read between two consecutive plot's update.

**Zoom freq min/max:** min and max of the frequency band shown on the Y axis (this does not prevent to make an unzoom once the plot is done and to see all the frequency band between 0 and the Nyquist frequency).

**zmin / zmax (color scale):** min and max values of the color scale which indicates the transfer function module in the time-frequency plot. You can improve the contrast by reducing the span between zmin and zmax. If  $zmin \geq zmax$ , then the option flag autoZ is automatically activated.

**logx:** put log scale on the x axis of the plot

**logy:** put log scale in the y axis of the plot

**logz:** put log scale in the z axis of the plot

**gridx:** show vertical grid lines along the x axis

**gridy:** show horizontal grid lines along the y axis

**gridz:** show grid lines along the z axis

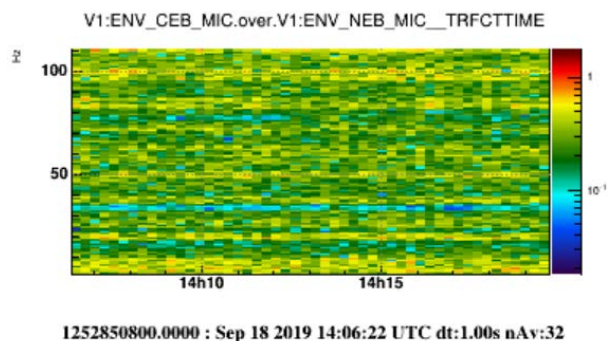
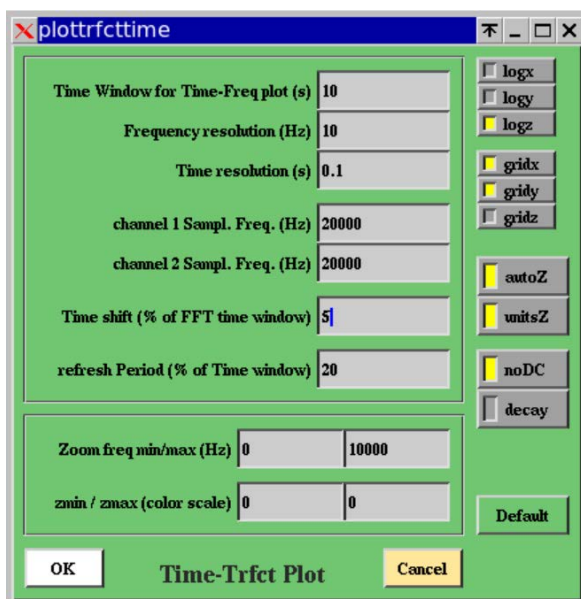
**autoZ:** sets automatically the min and max on Z axis according to the current data displayed

**unitsZ:** show on z axis calibrated data instead of raw ADC counts

**noDC:** subtract from the data the mean value computed over the data samples of the last FFT

**decay:** use a moving average when averaging the transfer functions

**Default:** put default values for all the parameters of the edit panel



## 5.10 COHE plot

The edit panel of the COHE plot has the following input fields and buttons:

**Time window:** number of seconds of data for the FFT used for the coherence computation

**Channel 1 Sampl Freq:** sampling or resampling frequency of the first channel

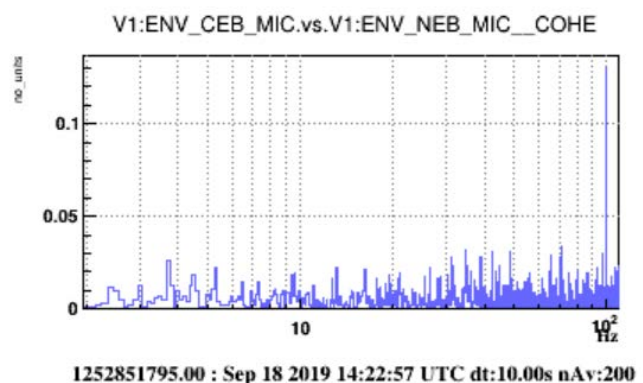
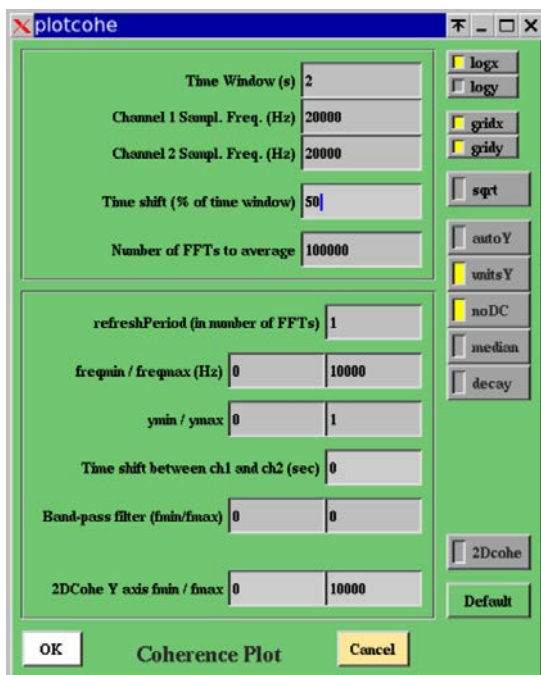
**Channel 2 Sampl Freq:** sampling or resampling frequency of the second channel

**Time Shift (% of time window):** shift over the input data between each FFT computation

**Number of FFT to average:** number of FFTs used to compute the coherence.

**Refresh period (in number of FFT):** number of FFT computed between each plot's update  
**freqmin/freqmax:** min and max of the frequency band shown in the plot  
**ymin/ymax:** min and max value of the coherence shown on the plot (should be between 0 and 1)  
**Time shift between ch1 and ch2:** shift (in sec) applied between channel1 and channel2 before computing the coherence between them  
**Band-pass filter (fmin/fmax):** min and max frequencies of the pass-band filter applied on the two channels before computing the coherence  
**2D Cohe Yaxis fmin/fmax:** min and max of the frequency band shown on Y axis in case a bicoherence plot is shown.

**logx:** put log scale on the x axis of the plot  
**logy:** put log scale in the y axis of the plot  
**gridx:** show vertical grid lines along the x axis  
**gridy:** show horizontal grid lines along the y axis  
**sqrt:** use correlation instead of the square of the correlation in the definition of the coherence  
**autoY:** sets automatically the min and max on Y axis according to the current data displayed  
**unitsY:** show on y axis calibrated data instead of raw ADC counts  
**noDC:** subtract from the data the mean value computed over the data samples of the last FFT  
**median:** compute a median coherence instead of an averaged coherence  
**decay:** use a moving average when computing the averaged coherence  
**2D Cohe:** if this flag is ON, a bicoherence plot is displayed instead of the usual coherence. The bicoherence allows to see coherence between channel1 at frequency f1 and channel2 at frequency f2  
**Default:** put default values for all the parameters of the edit panel



## 5.11 COHETIME plot

The edit panel of the COHETIME plot has the following input fields and buttons:

**Time window for Time-Freq plot:** number of seconds over which the time-frequency plot is shown  
**Frequency resolution:** width (in Hz) of each bin on the Y (frequency) axis



**Time resolution:** width (in sec) of each bin on the X (time) axis. If the time resolution is 10 sec and if each FFT is computed over 1 sec (frequency resolution of 1 Hz), the each bin on X axis will contain the average (or the median) over 10 FFTs.

**Channel 1 Sampl Freq :** sampling or resampling frequency of the numerator channel

**Channel 2 Sampl Freq:** sampling or resampling frequency of the denominator channel

**Time Shift (% of FFT time window):** time shift in input data between two consecutive FFT computation. It is a percentage of the FFT time window. The size of the FFT time window is the inverse of the frequency resolution. If the time shift is 30%, it means that the overlap between two consecutive FFT time windows is 70%.

**refreshPeriod (% of time window):** quantity of data read between two consecutive plot's update.

**Zoom freq min/max:** min and max of the frequency band shown on the Y axis (this does not prevent to make an unzoom once the plot is done and to see all the frequency band between 0 and the Nyquist frequency).

**zmin / zmax (color scale):** min and max values of the color scale which indicates the coherence in the time-frequency plot. You can improve the contrast by reducing the span between zmin and zmax. If zmin >= zmax, then the option flag autoZ is automatically activated.

**logx:** put log scale on the x axis of the plot

**logy:** put log scale in the y axis of the plot

**logz:** put log scale on the z axis of the plot

**gridx:** show vertical grid lines along the x axis

**gridy:** show horizontal grid lines along the y axis

**gridz:** show grid lines along the z axis

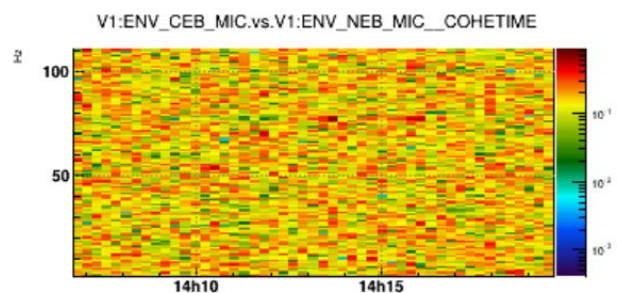
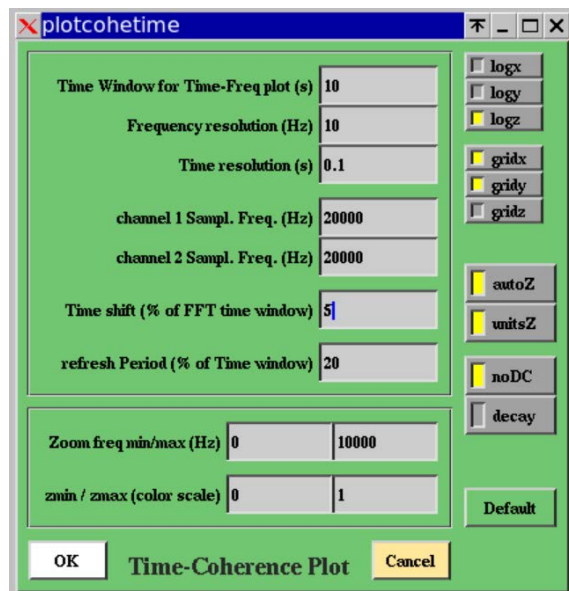
**autoZ:** sets automatically the min and max on Z axis according to the current data displayed

**unitsZ:** show on z axis calibrated data instead of raw ADC counts

**noDC:** subtract from the data the mean value computed over the data samples of the last FFT

**decay:** use a moving average when computing the average of coherences

**Default:** put default value for all the parameters of the edit panel



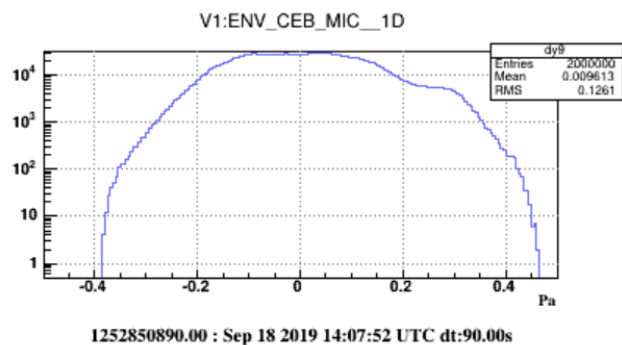
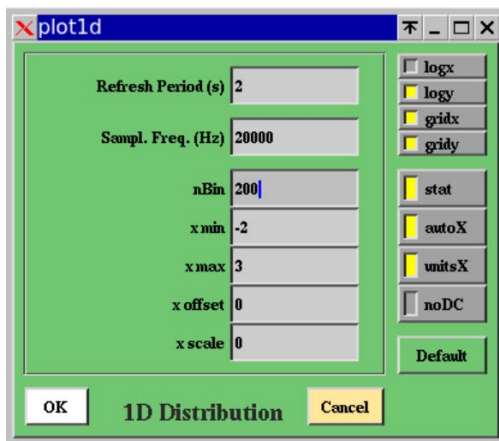
1252850816.0000 : Sep 18 2019 14:06:38 UTC dt:1.00s nAv:32

## 5.12 1D-DISTRIB plot

The edit panel of the 1D-DISTRIB plot has the following input fields and buttons:

**Refresh Period:** number of seconds of data read between two plot's update  
**Sampl. Freq. :** sampling or resampling frequency of the input channel  
**xBin:** number of bins initially used to show the 1D distribution of the input channel  
**xmin:** min value of the X axis  
**xmax:** max value of the X axis  
**x offset:** offset added to the input channel samples before computing the 1D distribution  
**x scale:** scale applied to the input channel samples before computing the 1D distribution

**logx:** put log scale on the x axis of the plot  
**logy:** put log scale in the y axis of the plot  
**gridx:** show vertical grid lines along the x axis  
**gridy:** show horizontal grid lines along the y axis  
**autoX:** sets automatically the min and max on X axis according to the current data displayed. In this case the xmin value is down extended by a factor 2 if one sample is below it and the xmax value is extended up by a factor 2 if one sample is above it.  
**unitsX:** show on x axis calibrated data instead of raw ADC counts  
**noDC:** subtract from the data the mean value computed over the data samples used for the distribution  
**Default:** put default values for all the parameters of the edit panel



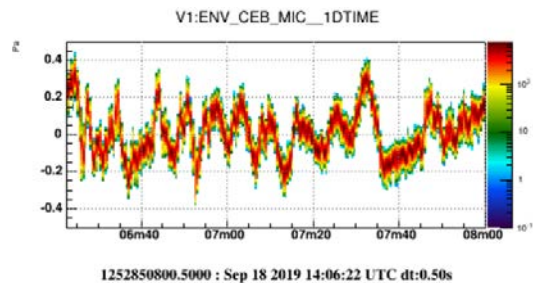
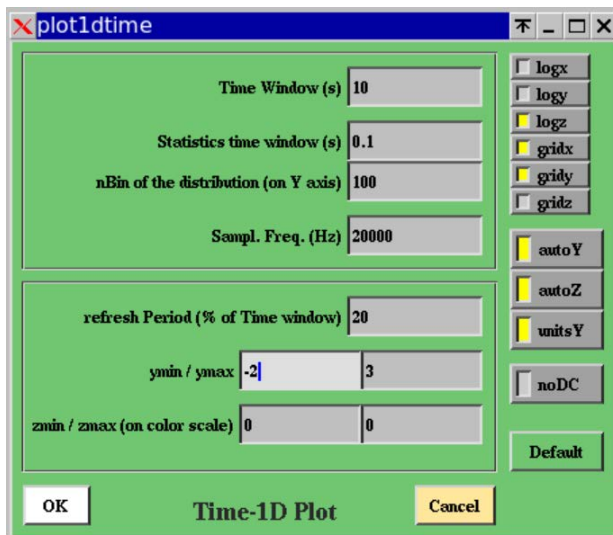
### 5.13 1D TIME plot

The edit panel of the 1D TIME plot has the following input fields and buttons:

**Time window:** number of seconds over which the 1D distribution evolution is shown  
**Statistics time window:** number of seconds of data used to compute one 1D distribution (thus number of bins on the X axis of the plot)  
**nBin of the distribution (on Y axis) :** number of bins used to shown the 1D distribution (thus number of bins on the Y axis of the plot)  
**Sampl Freq:** sampling or resampling frequency of the input channel  
**refresh Period (% of time window):** duration of the stretch of data read between two plot's update  
**ymin / ymax:** min and max of the 1D distributions shown on Y axis  
**zmin/zmax (on color scale):** min and max of the number of samples in the distributions

**logx:** put log scale on the x axis of the plot  
**logy:** put log scale in the y axis of the plot  
**logz:** put log scale in the z axis of the plot

**gridx**: show vertical grid lines along the x axis  
**gridy**: show horizontal grid lines along the y axis  
**gridz**: show grid lines along the z axis  
**autoY**: sets automatically the min and max on Y axis according to the current data displayed  
**autoZ**: sets automatically the min and max on Z axis according to the current data displayed  
**unitsY**: show on y axis calibrated data instead of raw ADC counts  
**noDC**: subtract from the data the mean value computed over the data samples used for the distribution  
**Default**: put default values for all the parameters of the edit panel



## 5.14 2D-DISTRIB plot

The edit panel of the 2D-DISTRIB plot has the following input fields and buttons:

**Refresh Period**: number of seconds of data read between two plot's update  
**Channel 1 Samp Freq**: sampling or resampling frequency of the first channel  
**Channel 2 Samp Freq**: sampling or resampling frequency of the second channel  
**xnBin**: number of bins on X axis, for channel 1  
**xmin**: min of the span shown on X axis for channel 1  
**xmax**: max of the span shown on X axis for channel 1  
**xscale**: scale applied on the samples of channel 1  
**xoffset**: offset added to the samples of channel 1  
**ynBin**: number of bins on Y axis, for channel 2  
**ymin**: min of the span shown on Y axis for channel 2  
**ymax**: max of the span shown on Y axis for channel 2  
**yscale**: scale applied on the samples of channel 2  
**yoffset**: offset added to the samples of channel 2  
**Time shift between ch1 and ch2**: shift (in sec) applied between channel1 and channel2 before displaying the 2D distribution, that is channel1 versus channel2.

**logx**: put log scale on the x axis of the plot  
**logy**: put log scale in the y axis of the plot  
**logz**: put log scale in the z axis of the plot (color scale)  
**gridx**: show vertical grid lines along the x axis



**gridy**: show horizontal grid lines along the y axis

**stat**: if this flag is ON, statistics box is displayed on the plot

**autoXY**: sets automatically the min and max on X and Y axis according to the current data displayed

**unitsXY**: show on X and Y axis the calibrated data instead of the raw ADC counts

**color**: display the plot with color scale on z axis

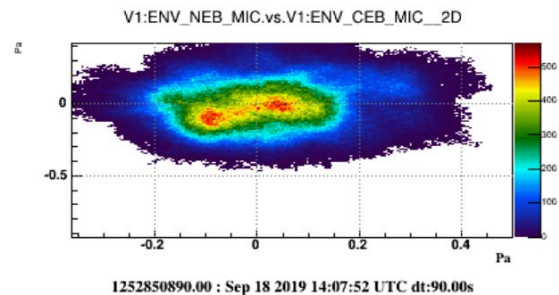
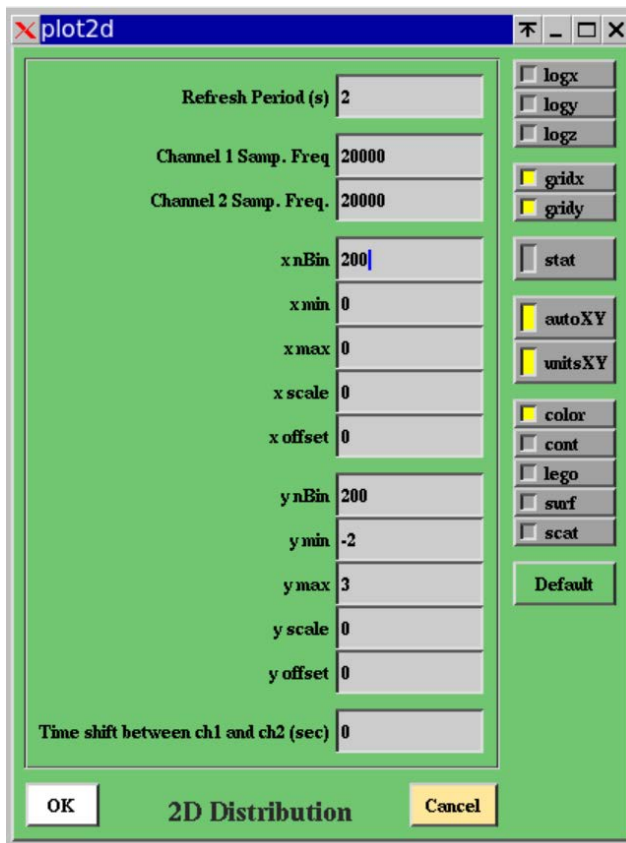
**cont**: display the plot with contours showing z-levels

**lego**: display the plot as a 3D lego plot

**surf**: display the plot as a 3D surface

**scat**: display the plot with points whose density is proportional to the value on z axis

**Default**: put default values for all parameters of the edit panel



## 5.15 RAW-IMAGE plot

The edit panel of the RAW plot has the following input fields and buttons:

**Sampl. Freq.** : sampling or resampling frequency of the input channel

**Number of images to average**: in case the input channel contains images, number of images to average between each plot's update

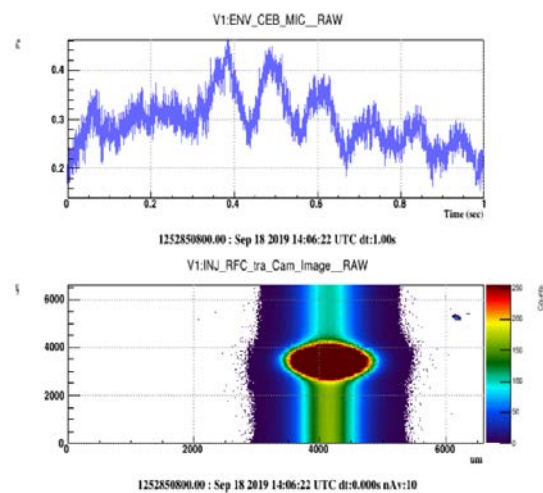
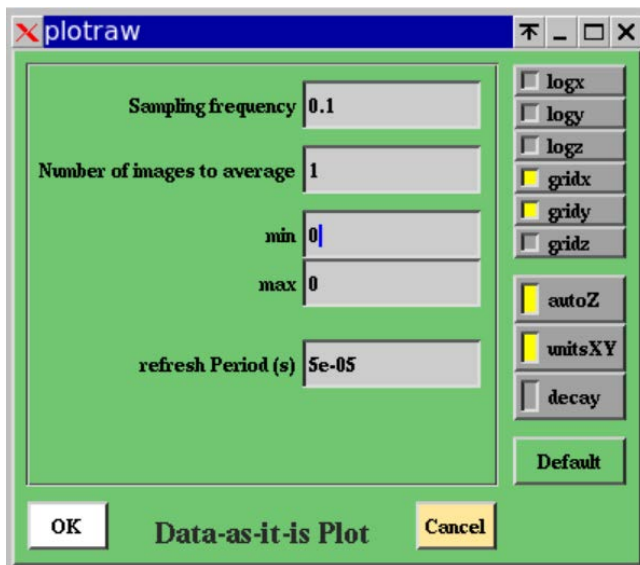
**min/max**: min and max values of the channel's samples displayed versus the parameter of X axis (which is time if the raw channel contains a time-serie, frequency if the raw channel contains a spectrum, etc...)

**Refresh Period**: number of seconds of data read between two plot's update

**logx**: put log scale on the x axis of the plot

**logy**: put log scale in the y axis of the plot

**logz**: put log scale in the z axis of the plot (if raw channel contains a camera image)  
**gridx**: show vertical grid lines along the x axis  
**gridy**: show horizontal grid lines along the y axis  
**gridz**: show grid lines along the z axis  
**autoZ**: sets automatically the min and max on Z axis according to the current image data displayed  
**unitsXY**: show on x and y axis calibrated data instead of raw ADC counts  
**noDC**: subtract from the data the mean value computed over the data samples of the last FFT  
**decay**: use a moving average when computing the averaged images  
**Default**: put default values for all the parameters of the edit panel

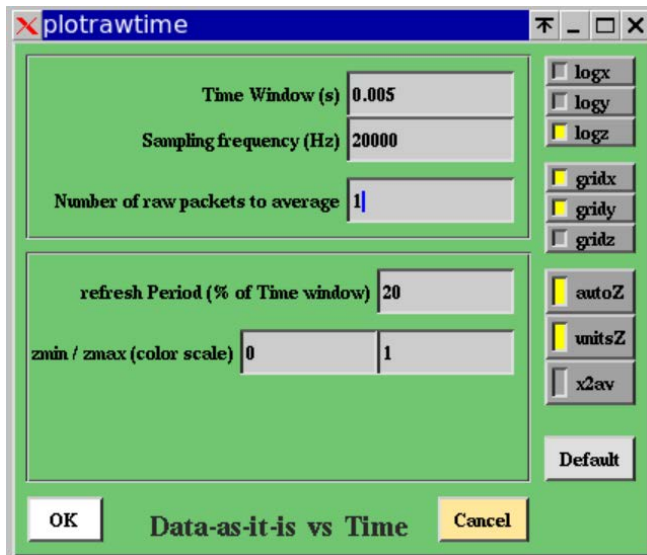


## 5.16 RAWTIME plot

The edit panel of the RAWTIME plot has the following input fields and buttons:

**Time window**: number of seconds over which is shown the evolution of the raw channel  
**Sampling frequency**: sampling or resampling frequency of the raw channel  
**Number of raw packets to average**: number of raw channel packets (one packet per frame) to average to fill one X bin of the plot  
**Refresh Period (% of time window)**: size of the data stretch to read between two plot's update  
**zmin / zmax**: min and max values of the raw channel to display

**logx**: put log scale on the x axis of the plot  
**logy**: put log scale in the y axis of the plot  
**logz**: put log scale in the z axis (color scale) of the plot  
**gridx**: show vertical grid lines along the x axis  
**gridy**: show horizontal grid lines along the y axis  
**gridz**: show grid lines along the z axis  
**autoZ**: sets automatically the min and max on Z axis according to the current data displayed  
**unitsZ**: show on z axis the calibrated data of the raw channel instead of raw ADC counts  
**x2av**: average the square of the raw channel packets instead of the raw channel packets  
**Default**: put default values in all the parameters of the edit panel



## 5.17 AUDIO plot

The edit panel of the AUDIO plot has the following input fields and buttons:

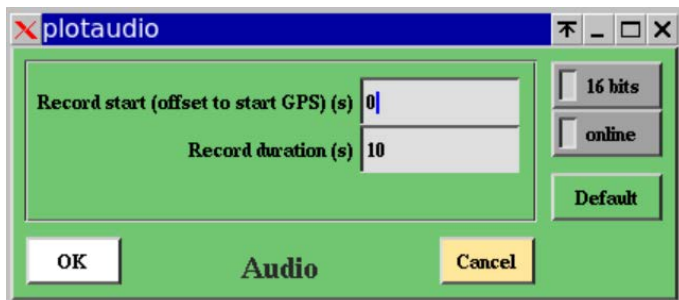
**Record start:** start GPS time of the data to read to be audio recorded

**Record duration:** duration of the data to read to be audio recorded

**16 bits:** if this flag is ON, the output audio channel will be on 16 bits instead of 8 bits (better dynamic)

**Online:** if this flag is ON, the output audio is sent to loudspeaker in addition to being recorded in a file

**Default:** put default values for all the parameters of the edit panel



# 6 Reference Plots

The button “Ref Plots” in the main panel shows a menu with the following items:

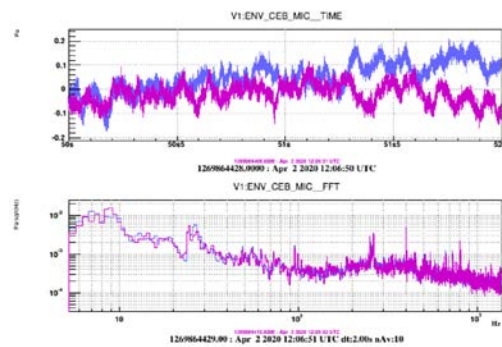
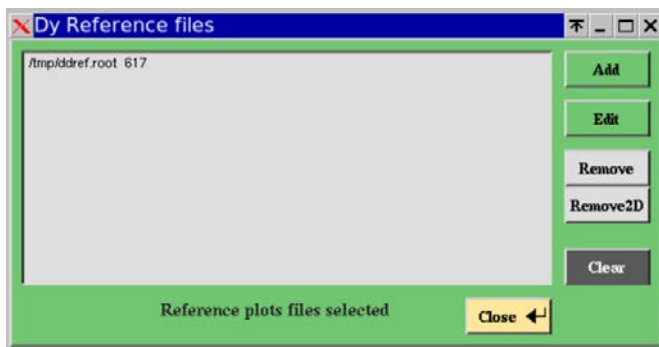
**Set Reference Plots:** Clicking on this button save the current plots into a file ddref.root and set them as reference plots to be superposed to the current plots. By default, the color of the superposed reference plots is purple.

**Remove Reference Plots:** Remove the reference plots currently superposed to the plots and remove the file ddref.root.

**Get Reference Plots:** Read from a file a set of reference plots to be superposed to the current plots.

**Save Reference Plots:** Save in a file the current reference plots

**Edit Reference Plots:** Set the color of the various reference plots that you previously get from various files. The edition panel of the Reference Plots allows to load reference plots taken from different files and to attribute a ROOT color code to each of them.



# 7 Tools and Options

The button “Tools” of the main panel shows a menu with the following items:

**Reset:** It allows to reset the list of plots and the list of channels as if you have never read data with the dataDisplay.

**Col x Row:** It allows to choose the number of columns and rows in the ROOT Canvas that displays the plots. This item is redundant with the “Plots Pattern” button available over the plots browser in the mainbrowser panel.

**Sliding along plots:** It opens a “Move in Plots” panel that provides several sliders to navigate inside the last plot selected or inside all the plots of the same type as the last plot selected (depending if the flag “all plots” is off or on). Let’s create 3 FFT plots. Now let’s zoom on the X axis of all of them. Then, you can use the Slider of the X position to move your zoomed view over the available frequency X axis. Item 4 is “Align Plots in time”. It changes time window and periods of updates so that all the plots are refreshed synchronously.

**Define User Functions:** It opens a panel with the list of user-defined functions that can be used to combine channels.

**Output data files:** It opens a panel where you can define a selection of channels to output, the directory where will be written the output frame files and the number of frames per file. Then, if you switch on the “Write frames on files” flag, the selected channels will be written on files each time the dataDisplay is running and reading data.

**Options:** It opens an Options panel where you can set various options about the channels browser, the plots browser and the display of the plots. Do one TIME plot and one FFTTIME plot, then try the various options to see their effect.

**Checks:** It outputs in the xterm window where the dataDisplay was started a full check list of the inputs, channels and plots parameters.

**Debug:** It opens a panel where you can choose the type of debug and the level of debug. You can also choose a waiting time between each plot update in order to see dataDisplay running at low speed.

## 7.1 The Options panel

The Options panel contains several options about the channels list of the plots display can be set off/on and where some parameter values can be changed for all the plots.

Below is the list of the options available:

**Log/Lin X:** Clicking on this button put log scale (alternatively linear scale) on the X axis of all the plots

**Log/Lin Y:** Clicking on this button put log scale (alternatively linear scale) on the Y axis of all the plots

**Zoom Legends :** Apply a new font size to all the legends of superposed plots (usually 0.07 is sufficient).

**Zoom Stat boxes :** Apply a new size of the stat boxes of all the 1D plots (usually 0.07 is sufficient).

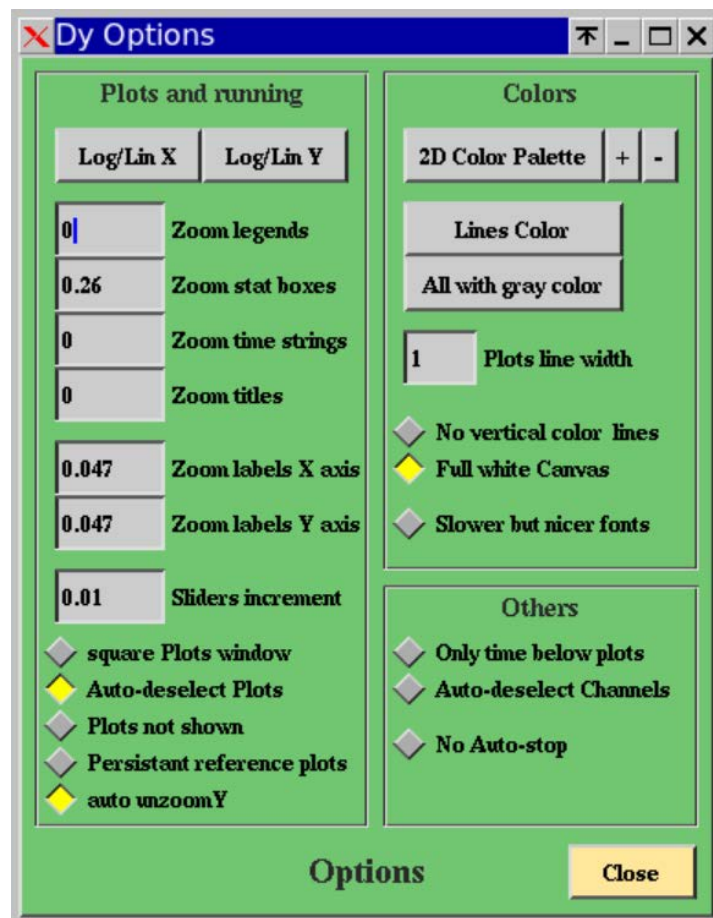
**Zoom time strings:** Apply a new font size to all the time strings shown under the plots.

**Zoom titles:** Apply a new font size to all the titles shown above the plots

**Zoom labels on X axis:** Apply a new font size to the labels on X axis of all the plots (usually 0.06 is ok)  
**Zoom labels on Y axis:** Apply a new font size to the labels on Y axis of all the plots (usually 0.06 is ok)  
**Sliders increment:** increment value used in Sliders panel. Default is 0.01.  
**Square Plots window:** if this flag is ON, the plots are forced to be square, whatever the canvas size.  
**Auto-deselect Plots :** if this flag is ON, plots in the plots browser are deselected after each action.  
**Plots not shown:** if this flag is ON, the plots are built but not shown until you click on Stop button  
**Persistent reference plots:** if this flag is ON, the reference plots are persistently superposed to plots  
**Auto unzoomY:** if this flag is ON, an automatic unzoom on Y axis is applied for any plot zoomed on X.

**2D Color Palette:** navigate in a predefined set of color palettes used for the 2-dim plots.  
**Lines Color:** each time you click on this button, a new color set is chosen for the 1-dim plots  
**All with gray color:** if this flag is ON, all the colors in 1-dim or 2-dim plots are turned into grey nuances.  
**Plots line width:** set the width of the lines in 1-dim plots. Default is 1. Better visibility is 2.

**No vertical colored lines:** if this flag is ON, the color bars indicating missing or corrupted data are removed. This removes also the color boxes that indicates ITF not locked or other user defined trigger.  
**Full white Canvas:** if this flag is ON, the ROOT canvas is fully white, without delimitation of each pad.  
**Slower but nicer fonts:** if this flag is ON, dataDisplay uses nicer fonts for all the labels and titles. But thos slows down the dataDisplay each time the plots are updated in the ROOT canvas.  
**Only time below plots:** if this flag is ON, no other information than GPS time and UTC date is shown.  
**Auto-deselect Channels:** if this flag is ON, channels in the ch. browser are deselected after each action.  
**No Auto-stop :** if this flag is ON, the dataDisplay will not stop automatically after 100 days of inactivity



Option panel

## 7.2 The Sliders panel

This panel allows moving the eyes position on a zoomed plot (or a full set of zoomed plots of the same type). Here are the following sliders:

**xposition**: the slider moves the position of the zoomed window on X axis of the last plot zoomed.

**xmin**: this slider change the min value of the zoom on the X axis of the last plot zoomed

**xmax**: this slider change the max value of the zoom on the X axis of the last plot zoomed

**yposition**: the slider moves the position of the zoomed window on Y axis of the last plot zoomed.

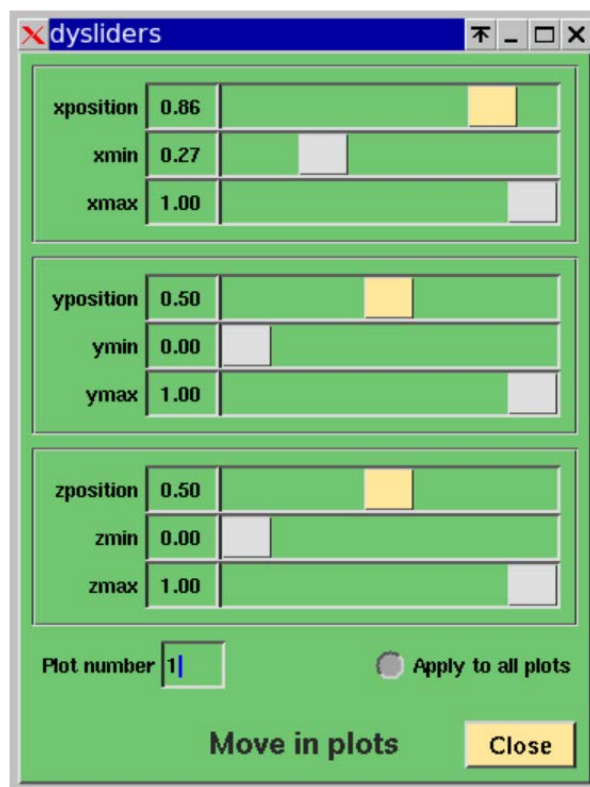
**ymin**: this slider change the min value of the zoom on the Y axis of the last plot zoomed

**ymax**: this slider change the max value of the zoom on the Y axis of the last plot zoomed

**zmin**: this slider change the min value of the zoom on the Z color axis of the last plot zoomed

**zmax**: this slider change the max value of the zoom on the Z color axis of the last plot zoomed

**Apply to all plots**: if this flag is ON, the sliders apply on all the plots of the ROOT canvas that are of the same type as the last zoomed plot.



Sliders panel



## 7.3 The debug panel

This panel allows choosing the part of the dataDisplay code to debug and the level of debugging. The following items of the panel are:

**General Check:** Write in the xterm window where the dataDisplay started a full check of the channels and plots parameters

**See Called functions:** if this flag is ON, the debugging will show the name of each function used by the dataDisplay when you use the GUI or when data are read and plots are computed and displayed.

**Speed monitoring:** if this flag is ON, the debugging will show the time (in milliseconds) passed in each step of the data reading and plots building steps, after you clicked on the “Start” button of the main panel.

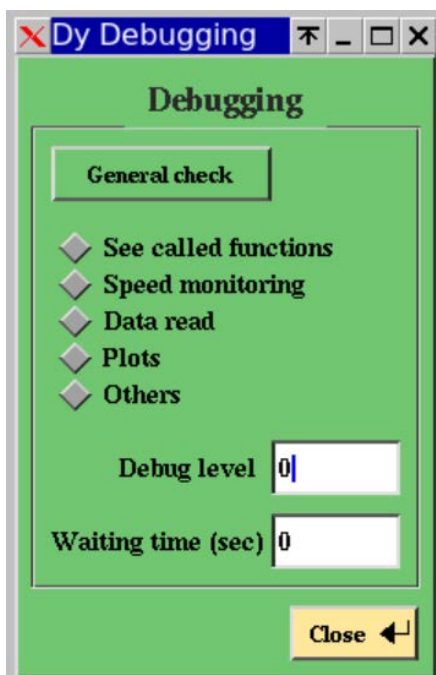
**Data read:** if this flag is ON, the debugging will show any debug message dedicated to the part of the code dedicated to connection to input sources and to data reading.

**Plots:** if this flag is ON, the debugging will show any debug message dedicated to the part of the code dedicated to plots generation, computation and display. It concerns also reference plots and action on plots like zooming.

**Others:** if this flag is ON, the debugging will show any debug message dedicated to the part of the code dedicated to any part of the dataDisplay code that is not directly dedicated to data reading or plots management.

**Debug level:** for each of the above debug types, the debug level 1 provides the main useful information. Level 2 provides a bit more details and level 3 is very verbose, with many details including the dump of each vector of input data.

**Waiting time:** This is the number of seconds between each plots update. The main purpose of this feature is to allow the user to have the time to see the changes in the plots between two updates.



Debug panel



# 8 Plots Root menus

The ROOT menus that appear when you right click on a plot or on the X or Y axis of a plot have additional features specific to dataDisplay. Those customized menus are described below.

## 8.1 The plot's line menu

If you put the mouse on the line of a 1-dimensional plot or on the bins of a 2-dimensional plot and right click, a ROOT menu appears. It contains some of the usual ROOT items but also additional items specific to dataDisplay:

<b>VDyPlot::dy1</b>	
ExportAscii	<b>ExportAscii:</b> Export in an ascii file the X,Y values of the plot. Such file can then be used for instance to do the same plot in matlab.
Scale_And_Offset	<b>Scale_And_Offset:</b> Apply a scaling factor to the data samples in the plot, and then apply an offset.
Scale_AroundMean	<b>Scale_AroundMean:</b> Apply a scaling factor to the data samples in the plot, after their mean value has been subtracted.
FFTOfDerivative	<b>FFTOfDerivative:</b> Compute the FFT of the derivative of the data samples in the plot.
FFTOfPrimitive	<b>FFTOfPrimitive:</b> Compute the FFT of the integral of the data samples in the plot.
Differentiate	<b>Differentiate:</b> Compute the derivative of the data samples in the plot.
IntegrateRms	<b>IntegrateRms:</b> Compute the integral of the rms of the data samples in the plot, starting from the last sample.
SuperposeExternalPlot	<b>SuperposeExternalPlot:</b> Take a plot from a ROOT file and superpose it to the current plot.
RemoveRefPlot	<b>RemoveRefPlot:</b> Remove any reference plot that may have been superposed to the current plot.
DrawPanel	
Fit	
FitPanel	
Rebin	
SetMaximum	
SetMinimum	
SetStats	
SetTitle	
SaveAs	
SetDrawOption	
SetLineAttributes	
SetFillAttributes	
SetMarkerAttributes	

## 8.2 The X axis and Y axis menu

If you put the mouse on the X axis of a plot, a ROOT menu appears. It contains some of the usual ROOT items dedicated to plots axis management but also additional items specific to dataDisplay:

TAxis::xaxis	
Log/Lin All	<b>Update Time/Freq:</b> Apply the same change of the width of the time window or the frequency window to all the plots (TBC).
Zoom All Labels	<b>Log/Lin All:</b> Put log scale (alternatively linear scale) on the X axis (resp. Y axis) of all the plots of the same type as the plot for which you right clicked on the X axis (resp. Y axis).
Zoom All Legends	<b>Zoom All Labels:</b> Zoom all the labels of the X axis (resp. Y axis) of all the plots if you right clicked on the X axis (resp. Y axis) of one plot.
Zoom All StatBox	<b>Zoom All Legends:</b> Zoom the legends of all the plots which are superposed or the title of all the plots which are not superposed.
Zoom ALL X as last	<b>Zoom All StatBox:</b> Zoom all the statistical boxes of the 1D plots.
Zoom X axis as last	<b>Zoom ALL X as last:</b> Apply to all the plots of the same type, the last zoom you did on the X axis of a plot.
UnZoom ALL X	<b>Zoom X axis as last:</b> Apply the last zoom you did on a X axis to the plot where you right clicked. This is canceled if the plots have not the same type.
UnZoomX	<b>UnZoom ALL X:</b> Unzoom the X axis of all the plots
Zoom ALL Y as last	<b>UnZoomX:</b> Unzoom the X axis of the plot where you right clicked.
Zoom Y axis as last	<b>Zoom ALL Y as last:</b> Apply to all the plots of the same type, the last zoom you did on the Y axis of a plot.
Scale Y as last	
Scale ALL Y as last	
UnZoom ALL Y	
UnZoomY	
UnZoom ALL Z	
UnZoom Z	
SetMoreLogLabels	
SetNoExponent	
SetDecimals	
SetRange	
SetRangeUser	
SetTicks	
SetTimeDisplay	
SetTimeFormat	
SaveAs	
SetNdivisions	

**Zoom Y axis as last:** Apply the last zoom you did on a Y axis to the plot where you right clicked. This is canceled if the plots have not the same type.

**Scale ALL Y as last:** Apply to all the plots of the same type, the last zoom you did on the Y axis of a plot, but using the relative scale of the zoom instead of the absolute values on Y axis.

**Scale Y as last:** Apply the last zoom you did on a Y axis to the plot where you right clicked, but using the relative scale of the zoom instead of the absolute values on Y axis. This operation is canceled if the plots have not the same type.

**UnZoom ALL Y:** Unzoom the Y axis of all the plots

**UnZoomY:** Unzoom the X axis of the plot where you right clicked

**UnZoom Z:** Unzoom the Z color axis of the 2-dim selected plot.

**UnZoom All Z:** Unzoom the Z color axis of all the 2-dim plots of the same type.

## 9 Useful Additional Tools

In addition to the dataDisplay software, the Dy package contains a set of useful tools to generate raw data frames for test, to view quickly raw data or to convert frame format to wav audio format and vice-versa.

### 9.1 rawframe.C

This ROOT script generates frames containing a predefined set of raw data channels, whose names are the following:

**vect1d** : a freq serie (a FFT) in FrProcData to be displayed with RAW-IMAGE  
**vect2d** : a 2D vector (image) in FrProcData to be displayed with RAW-IMAGE  
**vectbreakfreq** : a time serie in FrAdcData with a change of sampling frequency in the middle  
**vectbreakslope** : a time serie in FrAdcData with a change of adc slope in the middle  
**vectadc** : a time serie in FrAdcData with increasing value  
**vectproc** : a time serie in FrProcData with increasing value  
**vectproc2** : a time serie in FrProcData of type 2 with increasing value

All the channels contain a ramp, each sample having the value of its time indice, except vect2d which contains always gaussian noise added to the frame number.

**Syntax is:**

```
root -l rawframe.C\(\outputfilename, nframe, t0, framedt, noise\)
```

**Example :**

```
root -l rawframe.C\(\\"rawfile.gwf\", 30, 1112223334, 1., 1.\)
```

It generates 30 frames of 1 second, starting at GPS time 1112223334.

All the channels will contain a ramp added to a Gaussian noise of rms equal to 1.

Those frames are written in the output file rawfile.gwf

### 9.2 rawview.C

This ROOT script allows to view quickly the channels directly as they are stored in the frames. It is equivalent to using the plot RAW-IMAGE in the dataDisplay.

**Syntax is:**

```
root -l rawview.C\(\channel, deltat, inputfile, gpsstart, gpsend, ngroup, timesleep\)
```

**channel:** name of the input channel

**deltat** : the time step of data read

**inputfile:** the frame file from which channel is read

**gpsstart:** the starting GPS of the data read (if 0 then last data of the file are read)

**gpsend:** the end GPS of the data read (if below 700000000, it is then considered as a duration)

**ngroup:** number of samples regrouped before visualization (ngroup=0 means no regroup)

**timesleep:** the number of seconds between two successive updates of the plot

**Example 1: root -l rawview.C(\\"V1:SWEB\_B8\_Cam2\_Image\\",1,\\"/virgoData/ffl/raw\_O3b.ffl\\")**

It shows the camera image from SWEB\_B8\_Cam2\_Image. Data are read by bunches of 1 sec from the input file /virgoData/ffl/raw\_O3b.ffl

**Example 2: root -l rawview.C(\\"V1:Hrec\_hoft\_20000Hz\\")**

It shows the channel Hrec\_hoft\_20000Hz as it is stored in the frames. Default values of the other parameters are used (deltat=1, inputfile=raw.ffl, gpsstart=0, gpsend=20, ngroup=0, timesleep=1)

## 9.3 sound2frame

This compiled C code allows to convert a wave formatted audio file into a frame formatted file readable by the dataDisplay.

**Syntax is:**

/virgoApp/Dy/vXrY/\$UNAME/sound2frame.exe inputfile framelength

### *FRAME to WAVE*

If input file's name ends with .gwf or .ffl, it means that input file is a frames file.

Then sound2frame converts a frames file to a wave audio file whose name is frametowave.wav).

In this case, optional arguments are the channel name, the start GPS, the end GPS (or duration if less than 700000000)

**Example: sound2frame inputfile.gwf "ENV\_CEB\_MAG\*" 1270000010 100**

It read frame formatted data from inputfile.gwf , takes all the channels whose names start with ENV\_CEB\_MAG, over 100 sec starting at GPS=1270000010 and produces, for each channel, an output file channelname.wav

### *WAVE to FRAME*

If file's name ends not with .wav, it means that input file is an audio file.

Then, sound2frame converts a wave audio file to a frames file whose name is wavetoframe.gwf

In this case, optional argument is the frame length in seconds.

**Example: sound2frame inputfile.wav 10**

It reads the audio file samples and, for each bunch of 10 seconds, it writes a frame containing one channel named wavetoframe, in the output frames file named wavetoframe.gwf

If frame length is  $\leq 0$ , sound2frame converts a wave audio file to a wave audio file, named wavetowave.wav, and frame length is interpreted as a resampling frequency

**Example: sound2frame inputfile.wav -8000**

It read the audio file inputfile.wav, resamples the data at 8000 Hz and writes the result into the audio file wavetowave.wav

## 9.4 export2ascii.C

This ROOT script allows to export to an ascii file the content of a plot stored in a ROOT file.

The output ascii file has the following format: x\_value y\_value.

If the plot is data versus time, the x\_value is GPS time.

**Syntax is:**

```
root -l export2ascii\(filename, plotname, outname\)
```

**filename:** name of the input ROOT file that contains the plots

**plotname :** name of the plot. If you do not know it, put anything and you will get the list of plots

**outname:** name of the output ascii file. By default, outname is identical to filename with suffix “.txt”

**Example: root -l export2ascii.C\("Sensitivity\_20200322\_60Mpc.root", "hSensitivityH")**

It reads the content of the file Sensitivity\_20200322\_60Mpc.root, finds the plot hSensitivityH and output the plots data in an ascii file ./Sensitivity\_20200322\_60Mpc.txt

First 5 lines of the ascii file are (x\_value is frequency, y\_value is h sensitivity):

```
0.05 3.37247e-22
```

```
0.15 4.73475e-23
```

```
0.25 4.38881e-23
```

```
0.35 3.86777e-23
```

```
0.45 5.43065e-23
```

## 9.5 asciiview.C

This ROOT script allows to make a plot by reading an ascii file containing data in the following format:

x\_value y\_value

**Syntax is:**

```
root -l asciiview.C\("inputfile.txt", "mytttitle", "mytttitleX", "mytttitleY", logx_flag, logy_flag\)
```

**inputfile:** ascii file containing data x\_value y\_value

**title :** title of the plot (optional)

**titlex :** title of the plot X axis (optional)

**titley :** title of the plot Y axis (optional)

**logx\_flag :** If this flag is 1, the plot's X axis will be in log scale

**logy\_flag :** If this flag is 1, the plot's Y axis will be in log scale

**Example: root -l asciiview.C\("deltat\_freemich.txt", "DeltaL in Free Michelson", "Freq (Hz)", "DeltaL (m)", 1, 1\)**

It shows in a plot the content of the ascii file deltat\_freemich.txt. The axis are in log scale and titled “Freq(Hz)” and “DeltaL (m)”. The title of the plot is “DeltaL in Free Michelson”.

# 10 Bibliography

- [1] dataDisplay : VIR-0237A-15 <https://tds.ego-gw.it/?content=3&r=11672>
- [2] The Frame Format: VIR-0067A-08 <https://tds.virgo-gw.eu/ql/?c=2080>
- [3] The Frame library (Fr): <https://lappweb.in2p3.fr/virgo/FrameL/>
- [4] The Frame Vector library (Frv): <https://lappweb.in2p3.fr/virgo/FrameL/>
- [5] The Xforms library (Xforms): <http://xforms-toolkit.org/>
- [6] The ROOT library (ROOT): <https://root.cern.ch>
- [7] The Cm library (Cm): <https://git.ligo.org/virgo/virgoapp/Cm>

# 11 Appendix A : Video tutorials

dataDisplay video tutorials are available from:

<https://lappweb.in2p3.fr/virgo/dataDisplay> and <https://scientists.virgo-gw.eu/DataAnalysis/vod/>

Below are the texts of those video tutorials.

## 11.1 Video 1: Starting dataDisplay

From a machine on the Virgo site, to start dataDisplay, you can open an xterm window and type « dataDisplay ». This should open the dataDisplay mainpanel that provides you with a set of fields and buttons that we are going to describe in the following video tutorials.

The Virgo dataDisplay is a software tool mainly based on ROOT and Xforms libraries for the visualization and for the Graphical User Interface and on Framelib and Frv library for what concerns the data access and the signal processing. dataDisplay is dedicated to online and offline visualization of various type of signal processing plots taking as input the various channels provided by the Virgo data acquisition system.

We will see in the following tutorials how to access to those channels, how to combine channels, how to create and edit plots and how to use ROOT dataDisplay specific features to do various actions on the plots.

## 11.2 Video 2: Inputs

First, you need to access the data. If you click on the Input button, you get a menu showing the various types of data that you can access. Read FFL item is dedicated to the access to any frame formatted data through what is called ffl files. The ffl files are text files containing lists of frame formatted data files. For instance, if you choose the raw.ffl file, you will have access to all the raw data channels recorded by Virgo during the last months.

Using the Read File item, you can access directly to one or several frame formatted files, without the need of using a ffl. This is mainly dedicated to additional data sets that is not a Virgo official data stream or that you have produced yourself. Note that if you select several files, they will be considered as independent files instead of a serie of files containing data to be concatenated.

Using the Read Online item, opens a new interface panel which shows the various online servers that your dataDisplay can access to get data directly from the DAQ, with a latency of a few seconds.

Once connected to an online server, you get the list of the channels currently available. You can then manage the channels to do plots the same way as when taking data from input files.

One caveat: you can read data from several input files in parallel, but you can not read data from input files at the same time as data from online servers.

### **11.3 Video 3: Mainbrowser : Channels management**

Once you got the list of channels, you can choose one or several channels by clicking on them. Then, any plot you choose will be created for those selected channels.

In order to help you to find the channels you are looking for, a search field and a filter field are available. The search field get channels corresponding to the first letters you type in. The filter field allows moving at first place in the channel list the channels corresponding to your selection. For instance:

V1:ENV\_\*\_NE\*

To get the result, you need to type the “return” key.

If the channel you are looking for is not in the list, this is maybe because it is not always present in the data stream (either it is a channel with a sampling frequency below 1 Hz or it is a channel produced intermittently). In this case, click on the Update button after providing a GPS time for which you suspect that the channel is present.

In addition, you can use the upper buttons to sort the channels by name or by frequency or by input file name.

The CH. INFO button provides in the xterm window where the dataDisplay was started a short description of each channel selected.

### **11.4 Video 4: Mainbrowser: Combine channels**

You can also create new channels by combining various channels using the usual operations (+,-,x,/) and using any function available in math C library. For instance, select the three following channels and click on “Combine Channels” button. Then, you can edit the Operation line. In this example, we change the name of the resulting channel to “rate\_of\_power”, we change the default + signs and we apply a sinus function on the first channel. Clicking on the “Create new channel” button, a channel named “zz\_rate\_of\_power” is created. You can then use it to do plots, like for any other channels. Once the new channel has been created and even once plots have been created using this new channel, you can change the Operation formula that defines this new channel. If you create a second channel and then select again the channel “zz\_rate\_of\_power” the last formula used is shown in the Operation line.

### **11.5 Video 5: Mainbrowser: Plots management**

To create a new plot, just select one or several channels in the channels browser and click on one of the middle buttons (TIME, FFT, etc...). For each selected channel, one plot will be created and will be visible in the plots browser. If you choose 2D, COHE, or TRFCT, two channels are required for such plots. They will be taken in alphabetic order within the list of selected channels.



The button TIME creates a plot showing samples versus time.

The button BRMSTIME creates a plot showing the bandrms (rms computing over a limited frequency band) of the channel versus time. The bandrms samples are provided with the same sampling frequency as the original channel.

The button FFT creates a plot showing the power spectral amplitude (Fast Fourier Transform with  $1/\sqrt{\text{Hz}}$  units) of the channel.

The button FFTTIME creates a plot showing the time–frequency map or spectrogram (time evolution of the FFT) of the channel.

The button TRFCT creates a plot showing the module and phase of the transfer function between two channels.

The button TRFCTTIME creates a plot showing the time evolution of the module of the transfer function.

The button COHE creates a plot showing the coherence (correlation normalized by the square of the auto-correlations) between two channels.

The button COHETIME creates a plot showing the time evolution of the coherence.

The button 1D-DISTRIB creates a plot showing the distribution of the channel samples

The button 1DTIME creates a plot showing the time evolution of the distribution

The button 2D-DISTRIB creates a plot showing one channel versus another one

The button RAW-IMAGE creates a plot showing the raw values of a channel, as it is stored in the frames. For channels named \*\_image, it shows the images provided by Virgo cameras.

The button RAWTIME creates a plot showing the time evolution of the raw channel. If the channel contains for instance a spectrum (data versus frequency) instead of a time series, RAWTIME will show the time evolution of a spectrum, a spectrogram.

The AUDIO button creates an audio file in raw wav format on 8 or 16 bits, containing the samples of the channel. So, no plot here but a wav audio file is created for each channel selected.

Once you have created your plots, you can edit their parameters, as we will see later, but you can also do some operations on them:

The “Move Up” and “Move Down” buttons allow you to change the ordering of the plots in the list, which is the ordering of the plots in the canvas that will appear when you click on “Start” on the main panel.

The “Superpose” button allows to superpose in the same pad all the plots (of the same type) selected in the plots list. You can cancel the superposition by selecting superposed channels and clicking on the “Unsuperpose” button.

The “Permute Var” buttons allow to permute the channels within any 2D, COHE or TRFCT plots.

The “Get channel” button allows you to select in the channels list all the channels involved in the selected plots.

The “Hide” button allows you to hide from computation and display the selected plots. The “Show” button puts them back again in the game.

Next videos will show you how to edit plot parameters and how to switch on-off features specific to each plot type.

## 11.6 Video 6: Edit Plot TIME

If you created a TIME plot for the channel ENV\_CEB\_MIC, you can then click on the “Edit Plot” button to modify its parameters. You get a panel with several parameter fields and a set of options on the right side.

You can first choose to resample the data at 1000 Hz. You can also choose to update the plot only when new data have been read on a duration of 10% of the time window displayed. You can then modify the lower and upper bounds ymin and ymax of the TIME plot. In this case, do not forget to switch off the “autoY” flag on the right side.

You can also add an offset to the data displayed and rescale them. For instance, if you put an offset of 2 and a scaling factor of 3, the data will be scaled by a factor 3 and then offset by 2.

Now, let's click on the OK button.

We can then choose a starting date and a duration, on the main panel and click on the "Start" button. We get a plot showing the data of the ENV\_CEB\_MIC channel, resampled at 1000 Hz, multiplied by 3 and offset by 2...

Now, let's go back into the TIME plot edition panel. You can also apply a pass-band filter to visualize only the part of the signal which is between the two frequencies  $f_{min}$  and  $f_{max}$ . Here, we choose the frequency band between 20 and 50 Hertz.

Now, let's click on the OK button.

We get a plot showing the data of the ENV\_CEB\_MIC channel, resampled at 1000 Hz, multiplied by 3 offset by 2 and filtered between 20 and 50 Hertz...

Now, let's go back into the TIME plot edition panel and put back to zero the offset and scale and put back  $f_{min}$  and  $f_{max}$  to zero.

If we switch ON the "Show bits" flag, we get a quite different plot. This time, all the bits of each sample is displayed instead of the value of the sample. This can be helpful when the displayed channel contains bit-coded information, like a Virgo data quality state vector. Let's do a time plot for the channel DQ\_ANALYSIS\_STATE\_VECTOR, edit it and switch ON the flag "Show bits". Click on OK and start data read. You see that the bits 1,2 and 4 of each sample of this channel are at zero.

## 11.7 Video 7: Edit Plot FFT

If you created a FFT plot for the channel ENV\_CEB\_MIC, you can then click on the "Edit Plot" button to modify its parameters. You get a panel with several parameter fields and a set of options on the right side. The main parameters are the size of the time window (number of seconds of data used to compute one FFT) and the number of FFTs to average. You can also choose to resample the data before doing the FFT. In this case an anti-aliasing filter is applied. You have the various usual options like the use of real units instead of ADC counts and the auto-scale on Y axis. In addition, you can also choose to compute a median FFT instead of an averaged FFT or use a running average instead of resetting the average every N FFTs. Here is the result of an FFT with running average over  $N=10$ ... Her is what you get with a reset of the averaging every 10 FFTs... If you want more information in the time string below the plot, you must deselect the "Only time below plot" button in the "Options" panel.

## 11.8 Video 8: Edit Plot TRFCT and COHE

If you created a TRFCT plot for the channels ENV\_CEB\_MIC and ENV\_NEB\_MIC, you can then click on the "Edit Plot" button to modify its parameters. You get a panel with several parameter fields and a set of options on the right side. First, you need to define the time window for the TF computation. You can also resample the data of each channel before computing the Transfer Function. As an option, you can add a delay between the two channels before computing the TF... You can also apply a pass-band filter on the channels before computing the TF... Same options as for the FFT plots are available in the right of the Edit panel. The resulting plot is in fact two plots in the same pad: one for the magnitude of the TF, one for the phase of the TF. The phase is shown between  $-\pi$  and  $+\pi$ .

Creating a COHE plot for the channels ENV\_CEB\_MIC and ENV\_NEB\_MIC, is similar to creating a TRFCT plot, except that the result is a single plot showing the coherence (cross-correlation normalized by auto-correlations of channels) between the two channels. When selected, the 2D\_Cohe button transforms the coherence plot into a bi-coherence plot that shows the coherence between channel1 at frequency  $freq1$  and channel2 at frequency  $freq2$ .

## 11.9 Video 9: Edit Plot FFTTIME, COHETIME, 1DTIME

If you created a FFTTIME plot for the channel ENV\_CEB\_MIC, you can then click on the “Edit Plot” button to modify its parameters. You get a panel with several parameter fields and a set of options on the right side. You should first take care of choosing the time window over which the spectrogram is shown. Then, depending on the time window value, you choose the time resolution and the frequency resolution, remembering that, for instance, a frequency resolution of 0.1 Hz requires a time resolution of 5 sec. In addition, if you choose to compute averaged FFTs, the number of averages must be included in the time resolution. The time shift can be decreased so that the time resolution is below the threshold required by the frequency resolution, but you should be aware that correlation between successive FFTs will then appear. A Help button is available in the FFTTIME edit panel that explains all these features. When selected, the button `medY` normalize each bin of the FFTTIME plot by the median value of the bins along the same row (same frequency). As you can see, this increases the contrast of any transient noise present in the time window of FFTTIME.

For a COHETIME plot, that shows the time evolution of the coherence between two channels, you have parameters similar to those of the FFTTIME plot, with the same constraints on the time resolution and frequency resolution. Just take care of choosing a time resolution sufficiently large so that the number of FFTs averaged is sufficiently high to provide a correct computed value for the coherence.

For the 1DTIME plot, that shows the time evolution of the 1D distribution of a channel, you have different parameters. The “Statistics time window” is equivalent to the time resolution of the FFTTIME plot. It is the time over which the distribution is computed at a given time bin. The “nBin of the distribution” governs the resolution over the Y axis, which is the axis of the distribution. If you choose a large value for nBin, take care that the Statistic time window is sufficiently large to have enough samples in the distribution for each time bin. By default, the `autoY` button is ON, but in several cases, you are advised to put it OFF and to put proper values for `ymin` and `ymax`. `Zmin` and `zmax` fields allows to see by default only part of the distribution vertical axis.

## 11.10 Virgo 10: Start/Stop/Pause and mainpanel infos

Once you have created your plots, you can set the start GPS time (or the start UTC date) and push the button “Start” to read the data. While data are read and plots are updated, you cannot change the plots parameters but you can modify the size of the plots canvas or zoom on the plots X and Y axis. At any time, you can push the “Pause” button to stop the data read. Then, you can use the “Next Refresh” button to see the next plots update. The “Continue” button allows to restart the continuous data read... The button “Wait for data” allows to read files which are currently written by the DAQ and to wait for the next data available before continuing the read. The button “Use trigger”, when selected, allows to use the condition defined in the main browser “Create new trigger” field. The plots are then updated only when this condition is fulfilled.... The field “Current sources” shows you all the input files or online connection from which are currently read the data. The bottom lines provides information about the last action done by the `dataDisplay`. To view all the past actions, just click on the button on the right of the bottom line.

## 11.11 Video 11: Ref Plots management

At any time, once the data reading is finished, you can either set the current plots as Reference plots (button `RefPlots` / Set Reference Plots). This automatically turns the current plots with a purple color... ).

Then, pushing the start button, you see the reference plots superposed to the current plots. If you do not need anymore those reference plots, you can remove them (button RefPlots/Remove Ref Plots). You can also superpose to the current plots a set of Reference plots already saved (button RefPlots / Get Reference Plots). You can also manage the Reference plots already saved (button RefPlots / Edit Reference Plots) and attribute to them a color.

## 11.12 Video 12 : Options

The “Tools” button on the main panel opens an items list.

Item 1 is “Reset”. It allows to reset the list of plots and the list of channels as if you have never read data with the dataDisplay.

Item 2 is “Col x Row”. It allows to choose the number of columns and rows in the ROOT Canvas that displays the plots. This item is redundant with the “Plots Pattern” button available over the plots browser in the mainbrowser panel.

Item 3 is “Sliding along plots”. It opens a “Move in Plots” panel that provides several sliders to navigate inside the last plot selected or inside all the plots of the same type as the last plot selected (depending if the flag “all plots” is off or on). Let’s create 3 FFT plots. Now let’s zoom on the X axis of all of them. Then, you can use the Slider of the X position to move your zoomed view over the available frequency X axis.

Item 4 is “Align Plots in time”. It changes time window and periods of updates so that all the plots are refreshed synchronously.

Item 5 is “Define User Functions”. It opens a panel with the list of user-defined functions that can be used to combine channels.

Item 6 is “Output data files”. It opens a panel where you can define a selection of channels to output, the directory where will be written the output frame files and the number of frames per file. Then, if you switch on the “Write frames on files” flag, the selected channels will be written on files each time the dataDisplay is running and reading data.

Item 7 is “Options”. It opens an Options panel where you can set various options about the channels browser, the plots browser and the display of the plots. Do one TIME plot and one FFTTIME plot, then try the various options to see their effect.

Item 8 is “Checks”. It outputs in the xterm window where the dataDisplay was started a full check list of the inputs, channels and plots parameters.

Item 9 is “Debug”. It opens a panel where you can choose the type of debug and the level of debug. There is also a “General Checks” button that provides the same functionality as the item 8. You can also choose a waiting time between each plot update in order to see dataDisplay running at a low speed.

Item 10 is “Help”. It opens the same Help panel as the Help button of the main panel.

## 11.13 Video 13 : Help and help ballons

Finally, in addition to those video tutorials and to the main dataDisplay documentation, you can get some first help by clicking on the Help button in the main panel, by reading the text of the Help panel. And switching ON the “Help ballons” flag. Then, whenever your mouse is on a GUI element (browser, button or input field), you get a short explanation about its purpose and about the way it works.

All the video tutorials are available in:

<https://lappweb.in2p3.fr/virgo/dataDisplay> and <https://scientists.virgo-gw.eu/DataAnalysis/vod/>

# 12 Appendix B : dataDisplay Configuration

Each time the Start button is pushed, the dataDisplay produces a configuration file dy.cfg which can be used when starting a dataDisplay. It contains a configuration which is a set of keywords followed by parameters. Below is a commented example of such a configuration.

## # The start GPS time and duration of the data to read

```
DY_TIMING
starttime 1268492700.000000 (Tue Mar 17 15:04:42 2020)
duration -1.000000
```

## # The various options of the plots display, plots browser, channels browser, data read, etc...

```
DY_OPTIONS
debug          0 # debug level
debugtype      1 # debug type (data read, plots, etc...)
noplot         0 # flag is 1 if plot visualization is not requested
squareplot     0 # flag is 1 if square plots are requested, whatever the ROOT Canvas config
autodeselectplot 1 # flag is 1 if plots are deselected in browser after edition
autodeselectch 0 # flag is 1 if channels are deselected in browser after use for plots
colorpalette   1 # Color palette used for 2D or time-frequency plots (default is 1)
waitdata       0 # flag is 1 if dataDisplay waits for new data once reached the end of file
fullwhite      1 # flag is 1 if ROOT Canvas and pads containing plots are all white
linewidth      1 # line width in the plots
no_autostop    0 # flag is 1 if dataDisplay stop is required once data have been all read
```

## # The inputs (files of shared memory or Cm connection) from which data are read.

```
DY_INPUT
chindex 0 V1: # file number in the input list and Prefix of the channel names
inputtype 0 GWF_FILE # file type (0=frame formatted file, 1=online frame via Cm, ...)
inputname /virgoData/ffl/trend.ffl # name of the input file
```

## # The configuration of the ROOT Canvas (placement of the plots)

```
DY_PADS
ncol 1 # number of columns in the ROOT Canvas
nrow 2 # number of rows in the ROOT Canvas (available pads for plots is ncol*nrow)
```

## # The parameters of the plot of the pad 1 of the ROOT Canvas

```
DY_PLOT 1 dy1 # Plot number and name of the plot
type 7 TIME # Plot type (7=TIME plot that shows data versus time)
```

```

numpad      1      # Pad number of the plot
superposed  0      # flag is 1 if this plot is superposed to first plot of the pad
hidden      0      # flag is 1 if this plot is kept hidden (not built, not displayed but kept in the list)
chx_name    V1:ENV_CEB_MIC      # name of the channel displayed
chx_inputname /virgoData/ffl/trend.ffl # name of the input file providing this channel
chx_inputtype 0      # Input type (0=frame file, 1=online frame via Cm, ...)
chx_sampFreq 1      # Sampling frequency of the channel
chx_resampFreq 1    # Resampling frequency of the channel
chx_realunits 1     # flag is 1 if the real units of the channel is used instead of ADC counts
gridx       1      # flag is 1 if grid is set on X axis of the plot
gridy       1      # flag is 1 if grid is set on Y axis of the plot
logx        0      # flag is 1 if log scale is set on X axis of the plot
logy        0      # flag is 1 if log scale is set on Y axis of the plot
ydb         0      # flag is 1 if ???

tsize       2e+06   # size of the time window displayed (in seconds)
tstep       10      # percentage of the time window filled at each step
ymin        2e+31   # lower value on Y axis (if autoscale not set)
ymax        -2e+31  # upper value on Y axis (if autoscale not set)
yoffset     0      # offset added to the values on Y axis
yscale      0      # scale applied to the samples of the signal displayed in the TIME plot
bits        0      # flag is 1 if bits of each sample are shown in the plot
nodc        0      # flag is 1 if average value (DC) is subtracted to the signal
autoscale   1      # min and max on Y axis are automatically set from what is displayed
filter_fmin 0      # lower frequency of the pass-band filter applied
filter_fmax 0      # upper frequency of the pass-band filter applied (no filter if fmin=fmax=0)

```

**# Trigger to exclude some data from pre-processing (filtering, resampling, etc...) and from the plots**

DY\_TRIGGERS

```

triggertype 0 # not used
triggerused  0 # flag is 1 if a trigger is used
triggerpause 0 # flag is 1 if Pause on data read is activated each time a trigger period stops

```

**# All the ROOT Canvas configuration is stored between the /\*< and >\*/ tags, as below**

```

/*<
{
//=====Macro generated from canvas: gDDCanvas/dy_verkindt_17976_plots
//===== (Mon May 18 18:55:45 2020) by ROOT version5.34/05

TCanvas *gDDCanvas = new TCanvas("gDDCanvas", "dy_verkindt_17976_plots",28,88,910,570);

gDDCanvas->ToggleEventStatus();

gDDCanvas->SetHighLightColor(2);

```

```

gDDCanvas->Range(0,0,1,1);
gDDCanvas->SetFillColor(0);
gDDCanvas->SetBorderMode(0);
gDDCanvas->SetBorderSize(2);
gDDCanvas->SetLeftMargin(0.15);
gDDCanvas->SetFrameBorderMode(0);
// ----->Primitives in pad: gDDCanvas_1
TPad *gDDCanvas_1 = new TPad("gDDCanvas_1", "gDDCanvas_1",0.005,0.005,0.995,0.935);
gDDCanvas_1->Draw();
gDDCanvas_1->SetNumber(1);
gDDCanvas_1->cd();
gDDCanvas_1->Range(0,0,1,1);
gDDCanvas_1->SetFillColor(0);
gDDCanvas_1->SetBorderMode(0);
gDDCanvas_1->SetBorderSize(2);
gDDCanvas_1->SetGridx();
gDDCanvas_1->SetGridy();
gDDCanvas_1->SetLeftMargin(0.15);
gDDCanvas_1->SetFrameBorderMode(0);
gDDCanvas_1->Modified();
gDDCanvas->cd();
gDDCanvas->Modified();
gDDCanvas->cd();
gDDCanvas->SetSelected(gDDCanvas);
}
>*/

```

## 13 Appendix C : Associated tools

A set of additional tools are provided within the dataDisplay package:

**sound2frame** : converts raw audio files into frame formatted files and vice-versa.

**scansrc**: to get all the calls to a given function inside the dataDisplay code (more for developer)

**dataSender**: It is a specific package, available in /virgoApp/Ds/v1r2. The dataSender can read data from files and send them on request to a dataDisplay, through the Internet network.

## 14 Appendix D : Needed libraries

dataDisplay is a C and C++ code which can be compiled under CentOS7 with **CMT** or **CMake** tool. The first version released that is described by this documentation is v10r10.

The dataDisplay package, as well as the required libraries can be downloaded from:

<https://lappweb.in2p3.fr/virgo/dataDisplay> and <https://git.ligo.org/virgo/virgoapp/Dy>

The dataDisplay package uses the following libraries:

**Frame library**: a Virgo/LIGO specific package to read, write and manage frame formatted data. This library is regularly maintained for the purposes of the Virgo and LIGO gravitational wave experiments.

**Frv library**: a Virgo specific package to manage data vectors, to apply basic operations (like band rms, rms or mean and median computations) and to apply various signal processing tools (like FFT, TF, filter....). This library is one of the basis of the data collection and online/offline signal processing in the Virgo experiment.

**Xforms library**: an old library to build easily a quite complete GUI, with buttons, browsers, etc... This library has kept a users mailing list and a low level of maintenance. But its robustness through SL5, SL6, SL7, CentOS6, CentOS7... has allowed to keep it working well.

**ROOT library**: a graphical library developed at CERN since 1997. Maintained by a quite large community, this library is not always intuitive and easy to use but is quite powerful to manipulate and manage various plots and graphics.

**Cm library**: an old Virgo specific package to exchange data through Ethernet (TCP/IP packets) between two servers. This library is the basis of the data collection in the Virgo data acquisition system and has been quite maintained within Virgo.

In addition, part of the code of the dataDisplay is a copy of the function of the **Fd library** used by the Virgo data acquisition system and dedicated to exchange frame formatted data through the Ethernet network.