CNRS
Centre Nationale de la Recherche Scientifique

INFN
Istituto Nazionale di Fisica Nucleare

VIRGO Data Display
Users's Manual

D. Verkindt, D. Buskulic

LAPP, phone: 00.334.50.09.16.00, email: verkindt@lapp.in2p3.fr

VIR-008A-08
Version: v9r12p5
Date :   April 29, 2009

Contents

1 Introduction

The output of the Virgo Online Acquisition is numerical data structured as Frames [**?**]. Those frames contain raw data (for instance ADC values) as well as processed data (h values, results of online monitoring or online analysis, ...). To view and plot such data, the VIRGO experiment uses the dataDisplay software described in this note.

The data Display gives the possibility to dump data, to hear the data, to plot the data, to select the data to be viewed, to edit plots and superpose them. It uses a simple Graphical User Interface (Xforms GUI) and the graphical capabilities of a standard tool used at CERN (ROOT):

- Xforms [2] is a light and quite easy GUI: a free software with many useful GUI objects and a convivial GUI designer.

- The graphical tools of ROOT [4] are used to view the plots. We have added specific features like multizoom, start GPS time, superposition of reference plots or export of the plots data in ASCII files.

Figure 1 shows an example of the plots made by the dataDisplay:
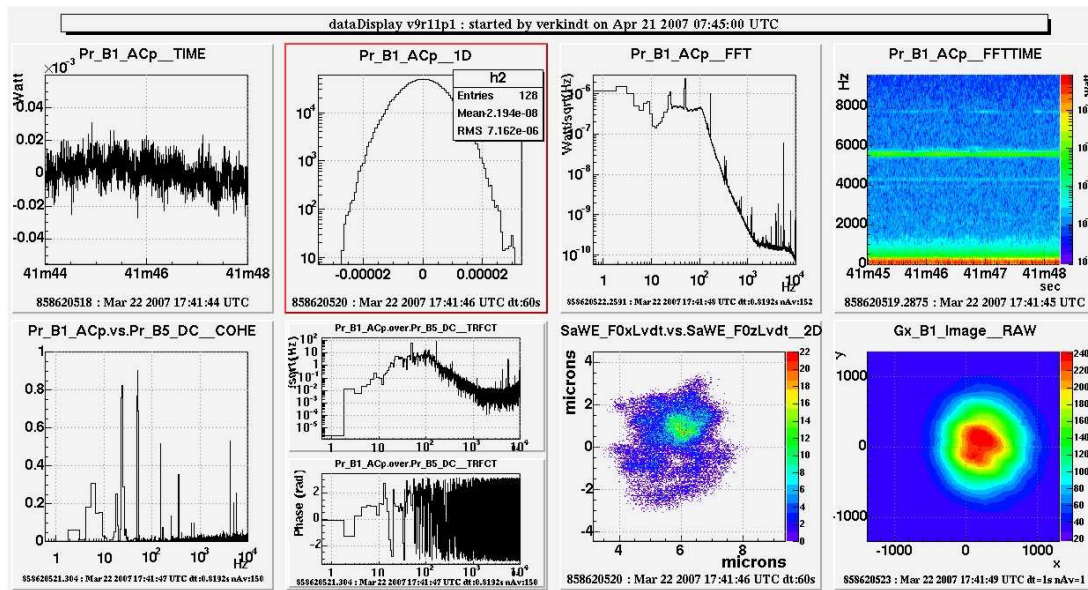
Figure 1: *Example of plots made by the Virgo Data Display.*

2 User's Manual

2.1 Starting dataDisplay

Command is just *dataDisplay*, once you got the Virgo environment. When starting dataDisplay, you can provide optional arguments:

Usage: dataDisplay [config] -f [start GPS] -l [duration] -o [filename] -t [title] -b -q -h -noplot -debug [dbglvl]

config = name of a configuration file.

- -f = next argument must be a GPS time in seconds (for instance 894368450) or an UTC date between quotes (for instance "9 May 2008 13:40:36").

- -l = next argument is the duration in seconds of data to be read.

- -o = next argument must be with extension .jpg or .root or .eps

- -t = next argument is the title that will appear on top of the plots canvas.

- -b = run in background mode without displaying plots, but producing a jpg file.

- -q = quit when stop is pressed or when all the required frames have been read.

- -h = to get the help message

- -f = followed by the GPS time where to start reading data

- -l = followed by the number of seconds of data to read

- -o = followed by the name of the output jpg file

- -noplot = plots are not displayed while running but a jpg is produced at the end

- -debug = followed by the level of debug

If [config] is provided, dataDisplay creates its *dd.car* file and starts immediately to read frames according to the configuration provided.

Each time the button "Start" is pushed, a configuration file *dd.car* is created in the directory where the dataDisplay was started. If you do not have write permission in this directory, *dd.car* is written in your home directory.

2.2 To be done in a next version

- Replace .. by _ in the Slow Monitoring channels names

- Fix the bug of one additional second of missing frames on TIME plots, when receiving frames from a dataSender

- In the menu of "Read Files", do not interpret a link to a directory as a file, when we click on it.

- Better test the TRIGGER functionality and simplify the user interface, especially for triggers lists in ASCII files.

- Add an anti-alias filtering when dataDisplay resamples the data

- Put in USER FCT and TRIGGER the possibility to use home-made functions (mean, rms, brms, filters...)

- Improve the way FFT of derivative or primitive is currently done.

- Replace the siesta-type main loop by a FdIO-type main loop. This would simplify the management of the plots, especially the superposition of signals with different sampling frequencies.

- Do the Draw of the plots only once for all or only just before the Canvas Update.

2.3   Bug fixes and new functionalities between v9r12p4 and v9r12p5

- Use the frame format v8 and frame library v8r01

- Fixed a time offset problem when reading trend data (time axis was correctly offset but not the data contained by the plot)

- Put a security against full frames received from a Cm server (number of channels in frame received must always be lower than twice the number of plots).

- Changed what is displayed by the coherence plot. Now what is shown is the module instead of the module*module.

- Fixed a bug in the conversion of GPS to UTC used under the plots (only visible when looking at data before 1st Jan 2009)

2.4   Bug fixes and new functionalities between v9r12p3 and v9r12p4

- Fixed a bug in the FFTOfPrimitive function of VDyPlot.cc

- Put units of transfer function module in the right order and removed sqrt(Hz)

- Fixed a bug in the zmin/zmax of the 2D RAW plots (camera images)

- Added an option to square the plots (useful to have the right dimensions of camera images)

- Fixed a bug in the use of Update Channels when connected online

- Fixed a bug on GPS time displayed below the plots, when doing a zoom and when using a Framelib version that has not the leap seconds of the data read.

- Added an item "Unzoom ALL Z" (can be convenient for camera images)

- Increased size of legends in the plots.

2.5   Bug fixes and new functionalities between v9r12p2 and v9r12p3

- Improve messages to the users when data are not available.

- Added a menu for connection to a shared memory.

- Fixed a bug in shared memory reconnection mechanism.

- Added possibility to update channels list when connected online (Cm) or when connected to a shared memory.

- Securized the code to receive frames online against UFrCurFrameH=NULL.

- Added possibility , when clicking on plot's line with right mouse button, to get FFT of Nth derivative or Nth primitive.

2.6   Bug fixes and new functionalities between v9r11p5 and v9r12 or v9r12p1 or v9r12p2

- Reduce size of the mainbrowser panel and reorganize buttons in it.

- Put white color in each pad. Added option to put white color also in the full ROOT canvas.

- Ask now for a password when connecting for the first time to a dataSender

- Fixed a bug in the connection to dataSender

- Simplified and improved the online Cm connection management

- Use now the function UHplotSetTime also for the 2D plots

- Fixed a bug in UHplotSetTime function and transfered this function into DyUtil.c

- Added a function UHplotUpdateEnv in URPlot.c

- Added filter for each channels browser of the mainbrowser panel. Typing "*Sc*Gc" make the browser show first all the channels whose name contains Sc and finishes with Gc.

- Added possibility of several colors for the Reference Plots. Choice of color can be done with a number next to file name.

- Added possibility to superpose reference plots while running (see "persistent ref. plots" button in the options panel)

- Added a panel to manage the reference plots files available in memory and get them from files

- Improved the "SuperposeExternalPlot" feature available when right clicking on a plot's line.

- Added a "RemoveRefPlot" feature in the menu available when right clicking on a plot's line.

- Added possibility to Hide temporarily some plots ("Hide" button next to plots browser)

- Added in Options panel the possibility to logx and logy on all plots, to change size of legends, stat boxes, titles, time strings, axis labels on all the plots.

- Fixed bugs in the automatic update when changing of input file data type (raw, 50Hz or trend data).

- Use of the ClXFSelector (selector of files) also for "Load Config" and "Get Reference Plot"

- Added a cursor bar in the mainpanel to move along data in the last zoomed plot

- Added a button "Transform" in the mainbrowser panel, that allows to change a plot from one type to an other.

- Fixed a bug in RAWTIME plots default parameters

- Added channels names on the axis label of plot 2D

- Delete the unused plots still in memory after each Stop.

- Improved and simplified the audio output capability (soundSim in URPlot.c).

- Added possibility to take also .wav audio files as input data.

- Added in the Dy package a utility sound2frame that converts .wav audio files into frames files and vice-versa.

- Improved the debugging messages. Use colors. Created the functions DyDebugXXX

- Added more comments and debug messages in URPlot.c

- Simplified the way of compiling under Cygwin (build_cygwin no more needed).

- Improved this documentation

2.7 Bug fixes and new functionalities between v9r11p4 and v9r11p5

- Fixed a bug on the time axis of the FFTTIME plots.

- Fixed a bug on the time axis of the RAWTIME plots.

- Fixed a bug in the search fields of the "data and plots" (mainbrowser) panel.

- Added possibility to ask remotely (through a Cm command) that dataDisplay put current plots into a jpg file.

- Added possibility to request ffl directory to the dataSender.

- Removed timeout on the frames reception through Cm

- Added possiblity to write on disk the frames received (button "Output" on main-panel).

2.8 Bug fixes and new functionalities between v9r11p1 and v9r11p4

- Put all options like "skipFrame", "noTOC", "Color2D", etc... into a specific panel accessible through the "Options" button of the mainpanel.

- Added possibility to save plots in ps or jpg format

- Added possibility to run the dataDisplay in background (and produce a jpg file)

- Added possibility to stop a dataDisplay remotely through a Cm command

- Several bug fixes and improvements that I do not remember...

2.9 Bug fixes and new functionalities between v9r10 and v9r11p1

- Fixed bug in the management of last data in the buffer for the TIME plots

- Fixed bug in the better management of missing frames in the TIME plots

- Fixed some bugs in the trigger functionality (but not yet fully tested)

- Button TRIGGER on mainbrowser panel now provides three choices:

  - Trigger from channels: you set a condition on some channels values. Then, the plots are updated only with data taken in the segments defined by this condition

  - Trigger from Ascii File: you provide the name of a file that contains a list of event's GPS times. Then, the plots will show only the data of the frames that contain those events. An automatic Pause is done by default between each event display.

  - Trigger from Event: you provide the name of a FrEvent (see in the procData browser all the names that begin with "EVENT") and a threshold (for instance: "EVENT_SNR ¿ 0"). Then, the plots will show only the data of the frames where the amplitude of the events is over the threshold. An automatic Pause is done by default between each event display.

2.10 Bug fixes and new functionalities between v9r09p2 and v9r10

- Coherence or Transfert functions between signals of different frequencies were done by resampling first the highest frequency signal. This could introduce aliasing. Now, the lowest frequency signal is oversampled but the plot is shown only on its meaningful part, below the lowest frequency/2.

- Now, output signal resulting from an operation on input channels (User Fct menu) are associated to different objects toto1,2,3,... of the same class toto.

- Slow Monitoring channels can now be used as inputs of Operations on channels (User Fct menu).

- Result of an Operation on input channels can be a logical one. For instance: $(Pr\_B1\_ACp < 0.01 \; AND \; Pr\_B5\_DC > 0.15) \; OR \; Qc\_Moni\_ScienceMode >= 1$

- A better management of missing frames in the TIME plots

- An option cohe2D in the Coherence plots editing panel allows to show a 2D plot with all the coherence coefficients. This allows to see cross-coherence between different frequency domains.

- Some bugs fixed in the edition of plots

- fixed bugs in the automatic plots changes when changing of input data (raw, 50Hz, trend)

- Resampling of data can now be done in the 1D plot.

- A button color2D has been added in the main panel. It allows to choose among 6 different color palettes for the 2D, FFTTIME, RAW (cameras images), COHETIME, TRFCTTIME, 1DTIME plots.

- Pause and Continue buttons can now be used when connected to the dataSender.

- A Trigger can be defined (Button TRIGGER on mainbrowser panel). It can be used (button Trigger pushed on mainpanel) and, optionally, there is an automatic Pause of the dataDisplay at each event encountered. This TRIGGER still needs some tests.

2.11   Bug fixes and new functionalities between v9r09p1 and v9r09p2

- Fixed wrong use of DyDecodeVar in DyFiles.c.

- Connection to a shared memory is now based on Fd/v5r0 code (see DyFdShm.c)

- Fixed a bug in DyGetDefaultFromPlot function of DyPlot.c

- Fixed a bug in conversion of date to GPS time (month name without upper case was not seen).

- Now, the plots configuration (nColumns x nRow) is saved into the configuration dd.car.

2.12 Bug fixes and new functionalities between v9r09 and v9r09p1

- No more file dd.root saved on disk.

- Button "Save" near plots browser saves now in a Root file only the displayed plots.

- Time delay reduced between pushing the "Stop" button and beginning to work on the plots.

- No possibility now to select two ffl files or a ffl and a frames file. So no more crashes when changing input data source.

- Possibility to superpose plots with different sampling frequencies

- No more time axis problem when resampling trend data at low frequencies

- GPS time shown under plots is now correct when using online connection.

- Possibility to visualize SMS data even if their sampling frequency is lower than frame frequency.

- Possibility to visualize data even if they are resampled at a frequency lower than frame frequency.

- Automatic update of plots parameters when changing the startGPS time

- Automatic update of plots and channels names when changing data source/type (raw data, 50Hz, trend data)

- Automatic deselect of channels or plots in the browsers (when button "Auto-deselect" is activated)

- Default parameters of a new plot are aligned on those of the plots of same type already present in the browser

- Possibility to do "Edit Clone" without changing parameters of a plot (just clicking on the round button next to parameter's field).

- Possibility to visualize bits of the signal shown on a TIME plot (button "Bits" in the TIME plot edit panel)

- Possibility to use "Pause" and "Continue" buttons when reading data from dataSender.

- Added TRFCTTIME and COHETIME plots that are time-frequency plots showing evolution in time of the transfert function (only the module) and of the coherence function.

- Added UnZoomX to solve the problem of superposed plot not fully displayed after a zoom-unzoom.

- Possibility to use ZoomAllXAsLast, ZoomAllYAsLast, UnZoomAllX or UnZoomAllY also with 2-dim plots.

- Waiting for an automatic tool, I have added a "Update Time-Freq" item in the Zooms menu that updates any frequency axis according to the last zoom made on a time-frequency plot.

- When using "GetRefPlots", the 2D plots are superimposed with their corresponding reference plots and any other 2-dim plot is replaced by the difference with its reference plot.

- When using "GetRefPlots", the 1D and 2D plots limits are updated automatically.

- UnZoomY does not take into account anymore the zeros at the end of TIME plot not completely filled.

- When channels is absent or corrupted (green or red vertical line on the plot) or when frame is absent (blue vertical line) the value under the colored line is not anymore 0 but is equal to the one of the previous bin.

- Possibility to compute min and max over N samples instead of resampling over N samples. For now, only implemented in TIME plot.

- Possibility to show plots only when a set of conditions on channels is fullfilled (trigger functionality). Implementation not yet finished.

2.13 Bug fixes and new functionalities from v9r09

- FFT, Transfert Function and Coherence can now be computed with a running average (button "decay" of the respective Edit panels).

- Time-frequency plot has a new Edit panel, where parameters "Time window for FFT computation" and "Number of FFTs to average" have been replaced by "time resolution" and "frequency resolution".

- Two new buttons in the mainpanel: "SetRefPlots" allows to save in a file the current plots as reference plots for a later usage" and "GetRefPlots" allows to superpose the reference plots to the current plots. In case of plots FFTTIME, 1DTIME and RAWTIME or RAW in two dimensions (camera images), superposition of plots is replaced by the difference between current plot and reference plot.

- Bug fixed in the last updated of a time plot when refresh period was over 50%

- In URPlot.c, added xxxIniFrame functions that do initialization for each plot when a first frame is read. This fix the bug of not receiving the units, slope and bias of the channels when getting data from the dataSender.

- Added GPS time origin under the 2D plots

- Fixed bugs for RAW, 1DTIME and RAWTIME plots for the units and for the automatic Y axis scale update.

- Put character "_" instead of characters ".." in the title of a plot, when the channel is a SMS data.

- Fixed many other small bugs like the one of button "nColxnRow" of mainbrowser panel...

2.14   Bug fixes and new functionalities from v9r08

- In addition to Fr, Xforms and Root, this dataDisplay needs only Cm and CSet to run online.

- Default start GPS time indicates now the most recent available data minus one hour.

- Default channels list is now made from the most recent available data minus one hour.

- Added a blue button near the "Start date (UTC)" field in the mainpanel. This button allows to get the "now" date.

2.15   Bug fixes and new functionalities from v9r06

- Possibility to UnZoom Y correctly all the plots superposed in the same Pad. This is available by clicking with right button of the mouse on the Y axis.

- Possibility to zoom the Labels on X or Y axis of all the plots. This is available by clicking with right button of the mouse on the plot's axis.

- Possibility to zoom the Legends of all the plots. This is available by clicking with right button of the mouse on the plot's axis.

- Possibility to multiply the plot by a scaling factor and to add an offset. This is available by clicking with right button of the mouse on the plot or, for TIME plots, by providing the scale and offset parameters in the edit panel.

- Possibility to superpose on the plot a "reference" taken from an other ROOT file. This is available by clicking with right button of the mouse on the plot.

- Macros containing the user's operations on the signals are kept now in the configuration file dd.car and also copied in /virgoData/DyAlgo.

- Possibility to do RAWTIME plots. This is useful especially when reading data which are already computed spectra and doing quickly a spectrogram.

- Possibility to get data from a dataSender which reads the data on disks and provide them to the dataDisplay through Internet. Need to do the command "cmdomain Cascina" before starting the dataDisplay.

- Bugs fixed in the use of the + and - buttons close to the plots browser.

- Possibility to change the plots distributions in the ROOT Canvas (button ColxRaw in the mainbrowser panel).

- Possibility to define a refresh period larger than the time window used to compute the FFT for Spectrum, Transfert Function or Coherence Function.

2.16   Bug fixes and new functionalities from v9r05

- Possibility to Scale as last on Y axis of a plot (Zoom Y as last but respecting the local scale of the plot).

- Possibility to zoom the Stat box of all the plots.

- Possibility to rescale a plot on Y axis (menu scale when clicking on the plot).

- Possibility to rescale a signal in a plot around its Y mean value.

- Possibility to superpose the differential on the plot.

- Possibility to superpose the integral summed from the end of the plot.

- Possibility to superpose the integral summed from the start the plot.

- Possibility to redo operations on signals without starting dataDisplay always from the same directory. Macros containing the operations on the signals are kept by default in /virgoData/DyAlgo.

- Possibility to superpose plots even for signals of different sampling frequencies.

- Possibility to do plots with signals that have sampling frequencies not submultiple of the fastest signal.

- Possibility to change the order of the plots, using the + and - buttons close to the plots browser.

- Possibility to visualize a 1D-time plot showing the evolution in time of the 1D distribution of a signal.

- Possibility to superpose the rms curve (integration of the spectrum starting at the Nyquist frequency) over the spectrum plot. (rms is not in units/sqrt(Hz)).

2.17   Bug fixes and new functionalities from v9r04

- Update of channels list (button "Update Channels") according to the GPS start value.

- Explicit use of ffl (frame files list) files to access huge amount of data.

- Possibility to visualize the raw values of a data vector (FrVect) of 1 or 2 dimensions. This allows to visualize Cameras images or to plot spectra data versus frequency that could be present in a FrProcData structure of a frame [?].

- Possibility to plot spectra in 1/Hz units instead of 1/sqrt(Hz).

2.18   New functionalities from v9r00

- Possibility to do operations on signals before doing plots (for instance, you will be able to do a plot on the variable "Pr_B1_ACq * Pr_B1_ACp / sin(Pr_B5_DC)". (blue button USERFCT above the proc data browser, and button USER FONCTIONS of the mainpanel).

- Possibility to propagate zoom on X or Y axis on all the plots

- Possibility to superpose several signals on the same plot (buttons SAME and UN-SAME near the plots browser)

- Possibility to exchange the two variables in a plot 2D, a plot TrFct (transfert function plot) or Cohe (coherence function plot). (button EXCHANGE near the plots browser)

- Possibility to visualize also Sim data, in addition to Adc, Proc and Sms data.

- Possibility to update the list of channels with respect to the GPS start time. (button UPDATE CHANNELS under the simdata browser)

- A predefined list of servers to connect to, when using dataDisplay online. (button PREDEFINED near the Cm servers browser).

- Date indicated in the mainpanel under the start GPS time is now in UTC and is updated directly when you change the start GPS time.

2.19   The main panel

The main panel (fig. 1) is made of the following buttons and fields:

- **Help**: A short help to start using the data display.

- **QUIT**: Quit data display if you answer "Yes" to the warning message.

- **Input**: To choose the dataDisplay's source of data (under frame format). **This is the first action to do**.

- **Configurations**: To save or load a configuration file. This configuration is for now a *.car file (siesta).

- **Data and Plots**: Once a first frame has been read from the source you selected, the adc data or slow monitoring data it contains are shown in the "Data and Plots" panel. Then, you can dump or plot those adc or sms channels.

- **Dump**: To see the current dump of the data you selected.

- **Output**: Provides access to the Output panel where can be set the options when writing on disk the frames read.

- **Cleaning**: Remove the graphical windows from the screen.

- **Reset All**: Reset all and put the data display in a "just started" state. Except the list of plots in the mainbrowser panel.

- **start Run# / Frame#**: To set run and frame numbers of the first frame to read.

- **start GPS time / duration**: To set the GPS time of the first frame to read. If used, then "Start Run# / Frame#" are put to -1 and 0 and are not used. Duration is the number of seconds of data over which you want the display to run once you have pushed the button "Start". Negative value means infinite.

- **start UTC date**: same as "start GPS time" but in explicit UTC date format. You can modify this field instead of "start GPS time" and the "start GPS time" field will be updated automatically.

- **Run# / Frame#**: run number and frame number of the current frame read.

- **GPS Time / Size**: GPS time and size (bytes) of the current frame read.

- **UTC date**: GPS time (in explicit UTC date format) of the current frame read.

- **SetRefPlots**: To save on disk a ROOT file that contain the current plots made.

- **GetRefPlots**: To superpose on the current plots, the Reference plots saved in a ROOT file.

- **EditRefPlots**: To manage (for instance change color of) the reference plots available in memory.

- **Data Source**: the current frames source (file, directory or a process accessed through the network). The first message on this line is "Give me input data...".

- **Start**: Start the data display. It creates the configuration file (a siesta cards file): "dd.car" and starts a Siesta like job, interactive with the data display GUI.

- **Stop**: Stop the read of data. A file "dd.hsto" containing the plot is created.

- **Pause**: Pause the read of data.

- **Next Refresh**: Continue to read data up to the next plot refresh.

- **Continue**: Continue to read data.

- **Next Frame**: Continue to read data up to the next frame.

- **Slider bar**: Run across the X axis, for all the plots last zoomed on X axis.

- **Options**: Provides access to the Options panel

- **Debug**: Provides access to the Debug panel

- **Help**: Check this button to get a short information about the dataDisplay.

- **Help Ballons**: Check this button to get online information about each button, browser or input field of each panel of the data display.

- **Info Line**: This line, at the bottom of the main panel, informs you about the last action made or the current status of the data display.
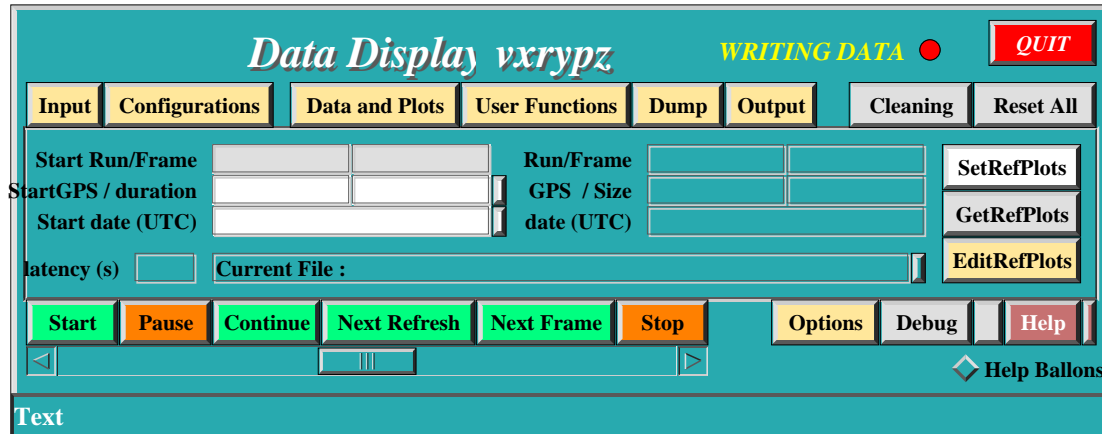
Figure 2: *Main panel of the Virgo Data Display.*

2.20 The options panel

- **Color2D**: this button allows to select predefined color tables for the 2D and time-frequency plots.

- **Log/Lin X**: put Log or Linear scale on the X axis of all the plots.

- **Log/Lin Y**: put Log or Linear scale on the Y axis of all the plots.

- **Zoom all legends**: Zoom in or out the legends of all the plots.

- **Zoom all stat boxes**: Zoom in or out the statistics boxes of all the plots.

- **Zoom all time strings**: Zoom in or out the time strings under all the plots.

- **Zoom all titles**: Zoom in or out the titles of all the plots.

- **Zoom all labels on X**: Zoom in or out the X axis labels of all the plots.

- **Zoom all labels on Y**: Zoom in or out the Y axis labels of all the plots.

- **Auto-deselect Plots**: selected lines in the plots browser are automatically deselected each time and Edit Plot panel is closed.

- **Auto-deselect Channels**: selected lines in the channels browsers are automatically deselected each time a plot is made.

- **no TOC**: on/off button to read frames without the files TOC (Table of Contents). In this case, reading of frames is slower but independent of any TOC failure.

- **Skip Missing Frames**: when selected, missing frames in the data stream read are not searched nor displayed (blue vertical lines) by the dataDisplay.

- **Plots not shown**: when selected, this button allows to make all the plots without viewing them. Once dataDisplay has been stopped, resulting plots can be find, as usual, in the file *dd.hsto*.

- **Persistant reference plots**: when selected, the reference plots in memory are superposed to the plots displayed.

- **No vertical colored lines**: when selected, no red, green or blue vertical lines in the plots appear anymore to indicate corrupted or missing data.

- **Full white Canvas**: when selected, All the canvas appears white like the pads.

- **Slower but nicer fonts**: when selected, the fonts used in all the plots are Helvetica (62) instead of Times (22).

- **Trigger ON**: when selected, the trigger function defined in the "data and plots" panel is used to selected the segments of data to be displayed.

- **Pause when trigger**: when selected, dataDisplay goes into Pause each time the trigger function defined in the "data and plots" panel has triggered.

2.21   The debug panel

- **Debug level**: level of debug information.

- **Checks**: Used only for debugging, so only for developpers. Close to this button, a white square allows to set the debug level.

- **Plots drawing**: Debug information concerning all the drawing/painting part of the code based on ROOT (mainly the code in VDyPlot.cc)

- **Plots building**: Debug information concerning all the data treatment and building of plots (mainly the code in URPlot.c)

- **Plots creation**: Debug information concerning all the creation and edition of the plots (mainly the code in DyPlot.c)

- **Inputs offline**: Debug information concerning all the data read from files (mainly the code in DyFiles.c)

- **Inputs online**: Debug information concerning all the data read online through Cm or shared memory (mainly the code in DyFiles.c, DyCm.c DyDd.c)

- **Data read**: Debug information concerning all the frames read (mainly the code in DyVar.c, DyFiles.c, UFrameL.c)

- **Channels management**: Debug information concerning all the channels management and the browsers of channels (mainly the code in DyVar.c, DyXform.c)

- **Configuration read/write**: Debug information concerning all the configuration read and write (mainly the code in DyConfig.c)

- **Speed of read/display**: Debug information concerning the time needed to read data or to display the plots (mainly the code in URPlot.c, DyFiles.c, DyUtil.c)

- **See called functions**: Show the name of the functions where the dataDisplay is currently passing

- **Any other stuff**: Debug information concerning various parts, mainly the GUI (mainly the code in DyUtil.c, DyXform.c dataDisplay.c)

2.22 Input Menu

This menu allows you to select the source of the frames (files or Cm server). among the following options:

- **Read FFL**: Take a set of ffl files (frame files list) as frames source. A file selector allows you to select several files in a raw, to search in directories (first line) and to put a tag on the files names (second line). Default tag is in this case "*.ffl". Each time you select this type of source, the files are added to an internal list of files that you can view and modify using the option "File List" of this "Input Menu".

- **Read File**: Take a set of files as frames source. A file selector allows you to select several files in a raw, to search in directories (first line) and to put a tag on the files names (second line). Each time you select this type of source, the files are added to an internal list of files that you can view and modify using the option "File List" of this "Input Menu".

- **Connect Online**: Take data online from a Cm server. If you select this menu item, you get the list of Cm servers available and you must choose one of them.

- **Connect to DataSender**: Take data online from a dataSender. If you select this menu item, you get a panel with by default a connection to the Cascina dataSender. You can then choose to connect to an other dataSender or choose the file you want to read through the dataSender. Clicking on a directory allows you to see its content.

- **Read Shared Memory**: Take data from a shared memory. If you select this menu item, you get an input field where you must provide the name of the producer of the shared memory to which you connect.

- **Read Directory**: Take data from all the files in a directory. Just select one file of the directory you want to add.

- **Read Files Online**: Take data from a directory where frames are currently written in files. There is a condition for using this option: all the files in the directory must have the same generic name containing an incremented number. When asked, just select one file of the directory you want to add.

- **Read Wave Files**: Take data from audio wave files. In this case, only one channel named "fraudio" is created.

- **Use fixed Cm Name**: Declare the data Display to the Cm Name Server. If you select this menu item, you are asked for a Cm name (default is your login name or the last Cm Name used).

- **Files List**: It opens a panel showing the list of input files that have been selected up to now and that are going to be read by the data Display once button "Start" has been pushed.

2.23   Configurations Menu

In this menu, you can "Load" a configuration, "Save" the present configuration with the name previously used or "Save as" the present configuration using an other name.

Each configuration file produced by the data Display is a *.car (Siesta cards) file. Each time you push the Start button, the current configuration is automatically saved in a file "dd.car". This is independant of what you do with the Config Menu items.

2.24   Data and Plots

This panel is the central part of the data Display. It shows buttons and browsers to select some frame's channels and to create plots.

2.24.1   The data browsers and buttons

- **Slow Monitoring Stations:** the list of slow monitoring stations (sms) contained in the frames. Selecting one gives you the list of data channels it contains.

- **SMS Data**: the list of channels of the selected sms. You can select one or several of them and push a plot button (TIME, 1D, FFT, ...). It will create a plot for each channel selected. To deselect a channel, click on it. To deselect all the channels, use the button "deselect all". A small input box under the browser allows you a fast search of a name in the browser.

- **ADC Data**: the list of frame's adc channels. You can select one or several of them and push a plot button (TIME, 1D, FFT, ...). It will create a plot for each channel selected. To deselect a channel, click on it. To deselect all the channels, use the button "deselect all". A small input box under the browser allows you a fast search of a name in the browser.

- **Processed Data**: the list of frame's proc channels. This is were you can find any processed (reconstructed) data. You can select one or several of them and push a plot button (TIME, 1D, FFT, ...). It will create a plot for each channel selected. To deselect a channel, click on it. To deselect all the channels, use the button "deselect all". A small input box under the browser allows you a fast search of a name in the browser. A "delete" button allows to delete new channels produced with the "User Fct" button.

- **Simulated Data**: the list of frame's sim channels. This is were you can find any simulated data. You can select one or several of them and push a plot button (TIME, 1D, FFT, ...). It will create a plot for each channel selected. To deselect a channel, click on it. To deselect all the channels, use the button "deselect all". A small input box under the browser allows you a fast search of a name in the browser.

- **Update Channels**: Changing the start date in the mainpanel and clicking on this button will update the channels lists according to the start date.

- **filter**: If you write "Em*BD*01" in this field and press Return, then the first items of the corresponding browser will be Em_ACBDCE01, Em_ACBDMC01, ..., Em_SEBDNE01, ... If you write "*", then all the channels will be presented in alphabetic order.

- **search**: If you write "Em*WE" in this field, the channel's name Em_ACBDWE01 will be found.

- **USER FCT**: This button provides a menu. The item "Operation String" of this menu allows the user to define its own operation on the signals and to create new proc signals that can be used to do plots. The other items are predefined functions that apply on the adc, sim or proc signals previously selected.

- **TRIGGER**: This button provides a menu. Each item of this menu allows to select a source of trigger condition. Each time a trigger condition is verified, the dataDisplay compute the plots and shows them, otherwise the plots are not computed. For instance a FFT can be computed with the condition that Qc_Moni_ScienceMode=1.

  - *Trigger from channels*: you can define a list of conditions on channels value, using the AND operator. For instance, "Qc_Moni_ScienceMode = 1 AND Pr_B5_DC ¡ 0.001 AND outlierMoni_Pr_B1_ACp ¿ 6"
  - *Trigger from ASCII file*: you provide the name of a file that contains a list of start GPS and duration or Start GPS and Stop GPS. Then, the dataDisplay will read data only in those time segments.
  - *Trigger from FrEvent*: you provide the name of an FrEvent channel and a condition. For instance "EVENT_SNR ¿ 10". Then, dataDisplay will search for frames where this condition is verified and will compute plots only with those frames.

2.24.2 The plots browsers and buttons

Clicking on one of those buttons creates a plot, if at least one data channel was selected.

- **TIME**: A time plot of each selected channel is created.

- **CORR**: A correlation plot between var2 and var1 is created.

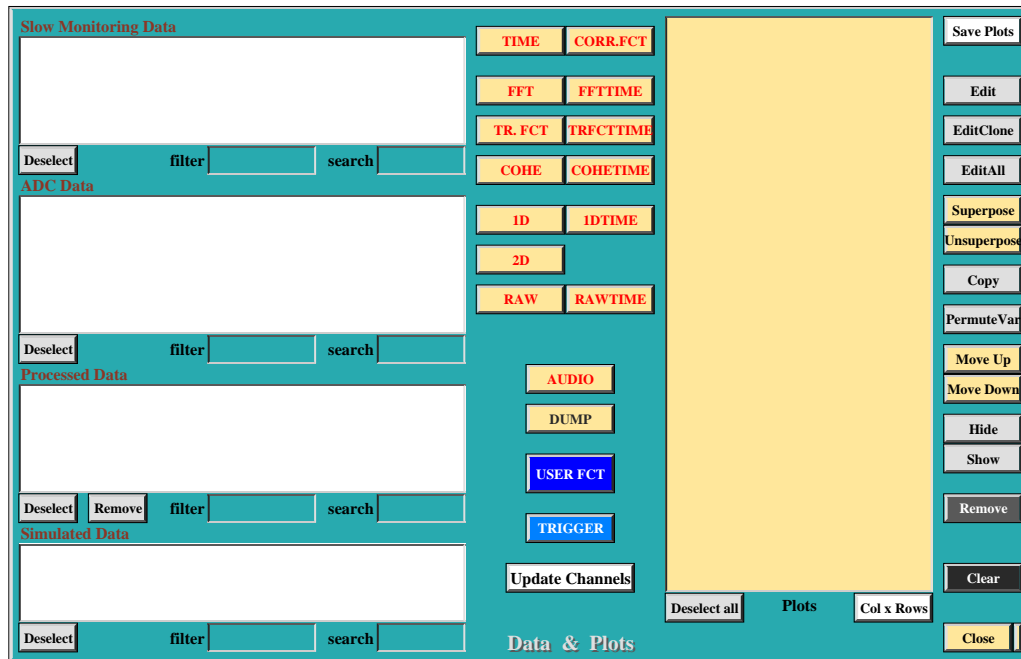- **FFT**: A spectrum plot of each selected channel is created.

Figure 3: *The "Data and Plots" panel of the Virgo Data Display.*

- **FFTTIME**: A time-frequency plot for each selected channel is created.

- **TR.FCT**: A transfer function plot between var2 and var1 is created.

- **TRFCTTIME**: A time-transfer function plot between var2 and var1 is created.

- **COHE**: A coherence plot between var2 and var1 is created.

- **COHETIME**: A time-coherence plot between var2 and var1 is created.

- **1D**: A distribution plot of each selected channel is created.

- **1DTIME**: A time-1D plot of each selected channel is created.

- **2D**: A 2d plot is created, showing var2 (y axis) versus var1 (x axis).

- **RAW**: A plot is created, showing raw values of a 1D or 2D data vector.

- **RAWTIME**: A plot is created, showing versus time raw values of a 1D data vector (for instance a spectrum).

- **AUDIO**: Selected channel can be hear over a given duration, after button Stop has been pressed.

- **DUMP**: Put the selected channels in the list of data channels to dump while reading frames.

Selecting one channel and pushing one of these buttons creates a plot (or audio or dump) with default parameters, that you can change using the "Edit" button of the plots browser. Selecting several channels of "SMS data", "ADC" or "Processed Data"

browsers and pushing one of those buttons creates as many plots (or audio or dump) as there are channels selected. If you push "2D", "TRFCT", "COHE", or "CORR" button, the selected channels are taken 2 by 2, from upper to lower. If a channel is pending, it is taken as Y variable, and the data display will take as X variable the channel you select just after.

- **Plots:** this browser shows the list of plots created. You can select one or several plots and edit or remove them.

- Save: this button allows to save the current plots according to the suffix you put in the name (root, ps or jpg).

- **Edit**: this button allows to change the parameters of the plots selected.

- **Edit Clone**: this button do the same as "Edit" but the parameters you changed are cloned to all the plots of the same type as the one you edited. This is convenient when you want to change for instance the time window size of all the time plots.

- **Edit All**: this button do the same as "Edit" but the parameters you changed are cloned to all the plots. This is convenient when you want to change for instance the time window size or the resampling frequency of all the plots.

- **Superpose**: this button put together (in the same root pad) the plots selected.

- **Unsuperpose**: this button separate (put in different root pad) the plots selected.

- **Copy**: create new plots by doing a copy of the plots selected.

- **Transform**: transform the selected plots into another type of plots, with default values and same channels.

- **PermuteVar**: this button exchange the variables of the 2D, TRFCT or COHE plots selected.

- **Move Up**: move up in the list all the plots selected.

- **Move Down**: move down in the list all the plots selected.

- **Hide**: this button puts in "hide state" all the plots selected. This means that they are still available but will not be displayed.

- **Show**: this button puts in "show state" all the plots selected. This means that they will be displayed.

- **Remove**: remove from the list all the plots selected.

- **Clear**: remove all the plots from the list.

Figure 4: *The TIME plot edit panel.*

2.25 The editing plot panels

To edit a plot, select it in the browser and click on "Edit". A panel is displayed, where you can edit the various parameters and plot options, depending on the plot's type. For more help about it, you can also use the Help Ballons button of the dataDisplay.

2.25.1 Edit panel of TIME plot

- **Name:** the name of the plot. By default, it is derived from the channels name. Do not put any blank in the plot name, this is not managed by Siesta cards.

- **Time Window:** number of seconds of data that are displayed in the time plot.

- **Sampling Freq.:** The new frequency at which you resample the data in order to view at larger scales the time evolution. This field contains by default the initial sampling frequency of the channel.

- **minmax:** If this button is selected, the resampling is done by computing the min and max values over 2N sampling instead of computing an average over N sampling.

- **Time Shift:** Percentage of the time window used to shift the time plot visualization.

- **ymin:** Minimal value of the time plot on vertical axis.

- **ymax:** Maximal value of the time plot on vertical axis.

- **Filter Band (fmin/fmax):** Min and max frequency of the pass-band filter applied on the data before time plot visualization.

- **logx:** on/off button to set log scale on horizontal axis.

- **logy:** on/off button to set log scale on vertical axis.

- **gridx:** on/off button to set grid lines along horizontal axis

- **gridy:** on/off button to set grid lines along vertical axis

- **autoY:** on/off button to have an automatic set of ymin and ymax, while plotting the data.

- **unitsY:** on/off button to set right units on the vertical axis.

- **noDC:** on/off button to set removal of mean value (DC component) of the signal.

- **Default:** Put the default values for all the fields and buttons.

- **OK:** Acknowledge the changes you made. Those changes will affect all the plots of the same type if you used "Edit Clone" instead of "Edit".

- **Cancel:** Cancel all the changes you made.

- **Small circles near each input field:** If such buttons are selected they indicate that EditClone should apply to those fields even if they have not been changed.

2.25.2 Edit panel of FFT plot

- **Name:** the name of the plot. By default, it is derived from the channels name. Do not put any blank in the plot name, this is not managed by Siesta cards.

- **Time Window:** number of seconds of data that are displayed in the time plot.

- **Sampling Freq.:** The new frequency at which you resample the data in order to view at larger scales the time evolution. This field contains by default the initial sampling frequency of the channel.

- **Time Shift:** Percentage of the time window used to shift the time plot visualization.

- **Number of FFT to average:** number of spectra you average. Visualization is done each time you add a spectrum, so you can see the progression of the averaging.

- **Refresh Period (in number of FFTs)**: the refresh of the plot's display is done every time this number of FFTs has been computed.

- **Zoom freq min:** Minimal frequency, just for visualization (no cut in frequency is really done).

- **Zoom freq min:** Minimal frequency, just for visualization (no frequency cut is really done).

- **Zoom freq max:** Maximal frequency, just for visualization.

- **logx:** on/off button to set log scale on horizontal axis.

- **logy:** on/off button to set log scale on vertical axis.

- **gridx:** on/off button to set grid lines along horizontal axis

- **gridy:** on/off button to set grid lines along vertical axis

- **autoY:** on/off button to set automatically the right vertical edges.

- **unitsY:** on/off button to set right units on the vertical axis.

- **noDC:** on/off button to set removal of mean value (DC component) of the signal.

- **1/Hz:** on/off button to set vertical axis in 1/Hz units instead of 1/sqrt(Hz).

- **rms:** on/off button to superpose the rms (integrate FFT starting from Nyquist frequency) on the FFT plot.

- **decay:** The averaging of FFTs uses a moving average.

- **Default:** Put the default values for all the fields or on/off buttons.

- **OK:** Acknowledge the changes you made. Those changes will affect all the plots of the same type if you used "Edit Clone" instead of "Edit".

- **Cancel:** Cancel all the changes you made.

2.25.3   Edit panel of FFTTIME plot

- **Name:** the name of the plot. By default, it is derived from the channels name. Do not put any blank in the plot name, this is not managed by Siesta cards.

- **Time Window for Time/Freq plot:** number of seconds of data that are displayed in the time-frequency plot.

- **Frequency resolution (Hz):** bin width on Y axis or the inverse of the number of seconds over which each FFT is computed.

- **Time resolution (s):** bin width on X axis or number of seconds of data between two consecutive averaged FFT computed.

- **Sampling Freq.:** The new frequency at which you resample the data in order to view at larger scales the time evolution. This field contains by default the initial sampling frequency of the channel.

- **Time Shift:** Percentage of the FFT time window used to shift in time before computing next FFT.

- **refresh Period:** Percentage of the time window used to shift along time (horizontal axis) the time-frequency plot visualization.

- **Zoom freq min:** Minimal frequency, just for visualization.

Figure 5: *The FFTTIME plot edit panel.*

- **Zoom freq max:** Maximal frequency, just for visualization.

- **Zmin:** Z value min threshold (used for color scale).

- **Zmax:** Z value max threshold (used for color scale).

- **logx:** on/off button to set log scale on horizontal axis.

- **logy:** on/off button to set log scale on vertical axis.

- **logz:** on/off button to set log scale on colors axis.

- **gridx:** on/off button to set grid lines along horizontal axis

- **gridy:** on/off button to set grid lines along vertical axis

- **gridz:** on/off button to set log scale on colors axis.

- **autoZ:** on/off button to set automatic computation of min and max on Z axis.

- **unitsZ:** on/off button to set right units on the colors axis.

- **noDC:** on/off button to set removal of mean value (DC component) of the signal.

- **decay:** The averaging of FFTs uses a moving average.

- **Default:** Put the default values for all the fields or on/off buttons.

- **OK:** Acknowledge the changes you made. Those changes will affect all the plots of the same type if you used "Edit Clone" instead of "Edit".

- **Cancel:** Cancel all the changes you made.

2.25.4   Edit panel of TRFCT or COHE plot

- **Name:** the name of the plot. By default, it is derived from the channels name. Do not put any blank in the plot name, this is not managed by Siesta cards.

- **Time Window:** number of seconds of data that are displayed in the time plot.

- **Sampling Freq.:** The new frequency at which you resample the data in order to view at larger scales the time evolution. This field contains by default the initial sampling frequency of the channel.

- **Time Shift:** Percentage of the time window used to shift the time plot visualization.

- **Number of FFT to average:** number of spectra you average. Visualization is done each time you add a spectrum, so you can see the progression of the averaging.

- **Refresh Period (in number of FFTs)**: the refresh of the plot's display is done every time this number of FFTs has been computed.

- **Zoom freq min:** Minimal frequency, just for visualization (no cut in frequency is really done).

- **Zoom freq max:** Maximal frequency, just for visualization.

- **logx:** on/off button to set log scale on horizontal axis.

- **logy:** on/off button to set log scale on vertical axis.

- **gridx:** on/off button to set grid lines along horizontal axis

- **gridy:** on/off button to set grid lines along vertical axis

- **autoY:** on/off button to set automatically the right vertical edges.

- **unitsY:** on/off button to set right units on the vertical axis.

- **noDC:** on/off button to set removal of mean value (DC component) of the signal.

- **decay:** The averaging of FFTs uses a moving average.

- **2Dcohe:** A bi-coherence plots (2D) is made instead of the usual coherence plot.

- **Default:** Put the default values for all the fields or on/off buttons.

- **OK:** Acknowledge the changes you made. Those changes will affect all the plots of the same type if you used "Edit Clone" instead of "Edit".

- **Cancel:** Cancel all the changes you made.

2.25.5   Edit panel of TRFCTTIME or COHETIME plot

- **Name:** the name of the plot. By default, it is derived from the channels name. Do not put any blank in the plot name, this is not managed by Siesta cards.

- **Time Window for Time/Freq plot:** number of seconds of data that are displayed in the time-frequency plot.

- **Frequency resolution (Hz):** bin width on Y axis or the inverse of the number of seconds over which each FFT is computed.

- **Time resolution (s):** bin width on X axis or number of seconds of data between two consecutive averaged FFT computed.

- **Sampling Freq.:** The new frequency at which you resample the data in order to view at larger scales the time evolution. This field contains by default the initial sampling frequency of the channel.

- **Time Shift:** Percentage of the FFT time window used to shift in time before computing next FFT.

- **Number of FFT to average:** number of spectra you average to produce one time slice of the time-frequency plot. Visualization is done each time a time slice is ready (average computed). Take care: if you choose for instance 10, the value real time window for plot visualization will be 10 times the value you put in the field "Time window for Time/Freq plot".

- **refresh Period:** Percentage of the time window used to shift along time (horizontal axis) the time-frequency plot visualization.

- **Zoom freq min:** Minimal frequency, just for visualization.

- **Zoom freq max:** Maximal frequency, just for visualization.

- **Zmin:** Z value min threshold (used for color scale).

- **Zmax:** Z value max threshold (used for color scale).

- **logx:** on/off button to set log scale on horizontal axis.

- **logy:** on/off button to set log scale on vertical axis.

- **logz:** on/off button to set log scale on colors axis.

- **gridx:** on/off button to set grid lines along horizontal axis

- **gridy:** on/off button to set grid lines along vertical axis

- **gridz:** on/off button to set log scale on colors axis.

- **autoZ:** on/off button to set automatic computation of min and max on Z axis.

- **unitsZ:** on/off button to set right units on the colors axis.

- **noDC:** on/off button to set removal of mean value (DC component) of the signal.

- **decay:** The averaging of FFTs uses a moving average.

- **Default:** Put the default values for all the fields or on/off buttons.

- **OK:** Acknowledge the changes you made. Those changes will affect all the plots of the same type if you used "Edit Clone" instead of "Edit".

- **Cancel:** Cancel all the changes you made.

2.25.6 Edit panel of 1D plot

- **Name:** the name of the plot. By default, it is derived from the channels name. Do not put any blank in the plot name, this is not managed by Siesta cards.

- **nBin:** number of seconds of data that are displayed in the time plot.

- **Refresh Period:** Number of seconds of data between two plot visualization refreshes.

- **x min:** Minimal value of the plot on horizontal axis.

- **x max:** Maximal value of the plot on horizontal axis.

- **logx:** on/off button to set log scale on horizontal axis.

- **logy:** on/off button to set log scale on vertical axis.

- **gridx:** on/off button to set grid lines along horizontal axis

- **gridy:** on/off button to set grid lines along vertical axis

- **stat:** on/off button to get statistics number on the plot.

- **autoX:** on/off button to have an automatic set of xmin and xmax, while plotting the data.

- **unitsX:** on/off button to set right units on the HORIZONTAL axis.

- **Default:** Put the default values for all the fields or on/off buttons.

- **OK:** Acknowledge the changes you made. Those changes will affect all the plots of the same type if you used "Edit Clone" instead of "Edit".

- **Cancel:** Cancel all the changes you made.

2.25.7 Edit panel of 1DTIME plot

- **Name:** the name of the plot. By default, it is derived from the channels name. Do not put any blank in the plot name, this is not managed by Siesta cards.

- **Time Window:** number of seconds of data that are displayed in the 1d-time plot.

- **Time Window for statistics:** number of seconds of data over which 1D distribution is done.

- **Sampling Freq.:** The new frequency at which you resample the data in order to view at larger scales the time evolution. This field contains by default the initial sampling frequency of the channel.

- **refresh Percent:** Percentage of the time window used to shift along time (horizontal axis) the time-frequency plot visualization.

- **nBin:** Number of bins of the 1D distribution (each vertical line of the plot)

- **Min:** Minimal value of the y axis (if autoY not selected).

- **Max:** Maximal value of the y axis (if autoY not selected).

- **Z max:** Z value threshold (used for color scale).

- **logx:** on/off button to set log scale on horizontal axis.

- **logy:** on/off button to set log scale on vertical axis.

- **logz:** on/off button to set log scale on colors axis.

- **gridx:** on/off button to set grid lines along horizontal axis

- **gridy:** on/off button to set grid lines along vertical axis

- **gridz:** on/off button to set log scale on colors axis.

- **noDC:** on/off button to set removal of mean value (DC component) of the signal.

- **unitsY:** on/off button to set right units on the Y axis.

- **Default:** Put the default values for all the fields or on/off buttons.

- **OK:** Acknowledge the changes you made. Those changes will affect all the plots of the same type if you used "Edit Clone" instead of "Edit".

- **Cancel:** Cancel all the changes you made.

2.25.8 Edit panel of 2D plot

- **Name:** the name of the plot. By default, it is derived from the channels name. Do not put any blank in the plot name, this is not managed by Siesta cards.

- **x nBin:** number of bins on horizontal axis.

- **x min:** Minimal value of the plot on horizontal axis.

- **x max:** Maximal value of the plot on horizontal axis.

- **y nBin:** number of bins on vertical axis.

- **y min:** Minimal value of the plot on vertical axis.

- **y max:** Maximal value of the plot on vertical axis.

- **Shift (n samplings):** shift in number of data samples between varX and varY. (varX can be identical to varY).

- **Refresh Period:** Number of seconds of data between two refreshes of the plot visualization.

- **stat:** on/off button to get statistics number on the plot.

- **logx:** on/off button to set log scale on horizontal axis.

- **logy:** on/off button to set log scale on vertical axis.

- **logz:** on/off button to set log scale on colors axis.

- **gridx:** on/off button to set grid lines along horizontal axis

- **gridy:** on/off button to set grid lines along vertical axis

- **gridz:** on/off button to set grid lines along colors axis

- **autoX:** on/off button to have an automatic set of xmin, xmax, ymin and ymax, while plotting the data.

- **unitsY:** on/off button to set right units on the horizontal and vertical axis.

- **color:** on/off button to use 2D plot with colors on third axis.

- **cont:** on/off button to use 2D plot with colored contours on third axis.

- **lego:** on/off button to use 3D plot with squares in each bin.

- **surf:** on/off button to use 3D plot with interpolated surface.

- **scat:** on/off button to use 2D plot with scattered points.

- **Default:** Put the default values for all the fields or on/off buttons.

- **OK:** Acknowledge the changes you made. Those changes will affect all the plots of the same type if you used "Edit Clone" instead of "Edit".

- **Cancel:** Cancel all the changes you made.

2.25.9 General informations about the plots

- **Time plot**: In the time plots, a blue vertical line is present each time at least one frame is missing, a green vertical line is present each time the channel is missing, a red line is present each time a data sample is corrupted (NaN, etc...).

- **FFT plot**: Units on the vertical axis are (channel units)/sqrt(Hz).

- **2D plot**: When title of the plot is var1_vs_var2, it means that var1 is on the vertical axis and var2 on the horizontal axis.

- **TrFct plot**: When title of the plot is var1_over_var2, it means that the Transfert Function plot is FFT(var1)/FFT(var2).

- **Cohe plot**: When title of the plot is var1_vs_var2, it means that the Coherence Function plot is FFT(var1*conj(var2))/(—FFT(var1)—*—FFT(var2)—).

2.26 User Functions Panel

In this panel, you can create or load a C or C++ function that will be then available in the "USER FCT" menu of the "Data and Plots" panel. A help button gives you some example on how to write such a function and what is not possible to write currently. The list of created or loaded functions are displayed in the browser. You can select one of them and compile it. But, it will be automatically compiled as soon as you need it to do operations on the channels.

2.27 Dump Panel

In this panel, the right browser shows the list of channels that will be dumped on the left browser. This list is increased by using the DUMP button of the "Data and Plots" panel. Button "Remove" allows to remove some of them. Button "Clear" removes all of them. Button "Full Frame" replace the dump of channels by a dump of the full frame received. The dump is done according to the dump level chosen. The content of the left browser can be saved in a file by using the button "Save". Dump information is visible only after the button "Start" of the mainpanel has been pushed.

2.28 Output Panel

In this panel, you can decide if you want to write on disk the frames received by the dataDisplay. At any time you can start and stop writing by checking or not the button "Write frames in file". You can choose a tag that defines the list of channels that will be written on disk, the name of the directory where will be written the frames and the number of frames per file. If the directory does not exist, it will be created at first frame's writing. The maximum number of bytes that you can write within a dataDisplay session is 10 GB. Writing of data is done as soon as the "Start" button of the mainpanel is

pushed, even if no plot is shown. If some plots are shown, the corresponding channels are automatically included in the data written on disk.

2.29   Five typical uses of the data Display

2.29.1   Read frames offline from a file

Suppose you would like to look at the Spectrum of a microphone signal. Data are under frame format in a file "toto.dat". First, in the main panel, select the item "Read File" of "Input Menu". Then, a Xforms predefined browser will ask you to select the file you want to read. The main browser panel is then displayed. You see in it a list of channels in the adc browser. Select the channel corresponding to the microphone and push the button FFT. The plot is made (with default parameters) as you can verify in the white bottom status line of the main panel. If you trust the default parameters, just push "Start" to read the frames and get the FFT plot. If you prefer to check or change the parameters, select your plot in the plots browser and push the button "Edit". A panel containing the name of the plot and its parameters is displayed. Change the parameters as you want and push "OK". Then, you can "Start". Once frames reading is finished or if you pushed the "Stop" button, the FFT plot has been recorded under PAW format in the file "dd.hsto".

2.29.2   Read frames online from files

Suppose you would like to look at the FFT of a microphone signal. Data are currently written under frame format as several frames per file in a directory "totoDir". You would like to look at the data while they are created. First, in the main panel, select "Read Files Online" from "Input Menu". A browser is displayed where you can select any file of the directory "totoDir". The main browser panel is then displayed. You see in it a list of channels in the adc browser. Select the adc corresponding to the microphone, and push the button "FFT". The plot is made (with default parameters) as you can verify in the white bottom status line of the main panel. If you trust the default parameters, just push "Start" to read the frames and get the FFT plot. The data Display will read the last frame of the files already present in the directory and will wait for the next file to be written in this directory, until you push the "Stop" button. For now, the data display is only able to read online files whose name is made of a generic name with an incremented number in it (for instance "FR12.dat","FR22.dat","FR32.dat" or "myframe_43","myframe_44"...).

2.29.3   Read frames online from a Cm server

Suppose you would like to get frames from a Cm server (a Frame Builder for instance). Choose "Connect to a Cm Server" in the "Input Menu". Select one Cm server in the list presented and double-click or click on OK. Then, list of channels is get or one frame is read and the "Data and Plots" panel should show the list of available channels. Then, once you created your plots, push the "Start" button. You should normally get frames from the Cm server and view the desired plots online.

2.29.4   Read frames online from a shared memory

Suppose you would like to get frames from a shared memory [8] (where a Siesta program is writing frames). Choose "Connect to Shm" in the "Input Menu". Provide the name

of the shared memory writer and click on OK. Then, the "Data and Plots" panel should show the list of available channels. Then, once you created your plots, push the "Start" button. You should normally get frames from the shared memory and view the desired plots at the rythm of the shared memory writer.

2.29.5 Read frames from the dataSender

Suppose you would like to get frames from a dataSender (that reads data from files and provides them through Ethernet). Choose "Connect to dataSender" in the "Input Menu". Choose a dataSender name and a file to read (or a directory to read) and click on OK. Then, the "Data and Plots" panel should show the list of available channels. Then, once you created your plots, push the "Start" button. You should normally get frames from the dataSender and view the desired plots at the rythm that can sustain the dataSender.

3 The Developer's Manual

3.1 Overview and Installation

The data Display version described here can be found in the /virgoApp/Dy directory, organized as the following:

- in *src, src/mysiesta, src/xform, src/userfct* directories: source code and header files.

- in *mgr* directory: the compilation Makefiles

- in *doc* directory: this documentation and some gif pictures showing dataDisplay.

"data Display" package uses ANSI C language compiled with cc and cxx under Unix OSF1 v5 (alpha workstations) platforms, and gcc under Linux platforms. It has been compiled under OSF1 (v4 or v5), under Linux (RedHat 7 or 9, SL4) and under Cygwin (so can be used within Windows environment).

It is based on the Xforms library for the GUI, on the Frame library for the data management and on the ROOT library for the plots display. The installation of dataDisplay needs at least:

- Xforms library: /virgoApp/Xform [2]

- ROOT library: /virgoApp/root [4]

- Frame library: /virgoApp/Fr [5]

For an online use of the dataDisplay, you need also:

- CSet package: /virgoApp/CSet [6]

- Cm package: /virgoApp/Cm [7]

For more informations about the packages used, you can have a look at the Web page: http://wwwcascina.virgo.infn.it/sDoc

Some modified source code of the siesta package [3] is also used. The source files are in the directory *mysiesta*: URPlot.c (replace UHistoL.c), UFrameL.c, UJobL.c, UArrayL.c, UMathL.c, USignalL.c and their header files, and c_hbook.h.

There are two global structures defined in dataDisplay.h: DyTool and DyPlots, and an external structure for each Xforms panel (see header files in src/xform directory).

3.2 Compilation scripts in directory mgr

In *mgr* directory, Makefiles dedicated to specific platforms are available:

- *Makefile.VIRGO.osf1.cxx6*: compile under OSF1, in Virgo environment.

• *Makefile.LAPP.linux.egcs*: compile under RedHat Linux v7 or v9 or SL4

The command is for instance *gmake -f Makefile.VIRGO.osf1.cxx6*.

There are also Makefiles with suffix "_nocm" (like Makefile.VIRGO.osf1.cxx6_nocm). They compile an offline version of the dataDisplay (uses only Frame, Root and Xforms libraries) that take input frames only from files or shared memory.

All the Makefiles contain an include of the file "Make-macros", which contains the common parts, "independant" of the platform, and an include of the file "Make-requirements" which contains the maximal list of the needed packages.

The mgr directory contains also a script named "makeUtil" to compile any additional tool (like sound2frame.c) provided in the Dy package.

3.3   Compilation with CMT in directory cmt

In *cmt* directory, thanks to JM Teuler and F. Carbognani, an updated requirements file is available. To use it, you need to have installed first the CMT package or to be already in the Virgo standard software environment. Under OSF1 or Linux, just go into the cmt directory and do the command *cmt config* followed by the command *gmake*.

To compile under Cygwin, you need first to uncomment some lines in the requirements file and to comment others (see instructions in the requirements file). A script named build_cygwin is also available in the cmt directory. It case of problem, it can used to do the link under Cygwin.

3.4   The main functions and structures

This data Display package uses two global C structures DyTool and DyPlot, pointed respectively by the pointers Dy and Plot[i].

• **DyTool**: It contains general parameters useful for most of the data display functions. For instance, Dy->Order is the status of the data Display (Stop, Pause, Start...). Each parameter of this structure is described in the header file dataDisplay.h.

• **DyPlot**: It contains the parameters of each plot, one DyPlot structure per plot, pointed by Plot[i]. Each parameter of this structure is described in the header file dataDisplay.h.

The two previous structures are used in almost all the functions. Here is a non exhaustive list of the data display functions (still to be updated!):

• **main**: Arguments can be passed to this function: the name of the configuration file, the number of frames to read and the debug level. In function main, the Xform library is initialized. Rest of the initialization is done by the function DyInit(), the panels are set and the main panel is displayed before entering the infinite *fl_do_forms()* GUI events loop.

- **DyReadFirstFrame**: It reads a first frame of the files list or from the Cm server and uses it to build the variables lists of the "Data and Plots" panel. This function is called each time you get a new configuration ("Config Menu") or a new source of frames ("Input menu").

- **DyMakeSiestaFile**: called when you save a configuration or most commonly by the DyStart function (that is called when you push the button "Start"). Using the structure *Plot, the data source, and the clocks list, DyMakeSiestaFile builds the configuration file (Siesta cards file) dd*.car that will be used when you push the "Start" button. It calls successively *DyMakeClockString, DyMakeFileString* and *DyMakePlotString*.

- **DySiestaLike**: called by DyStart, just after the call to DyMakeSiestaFile. It does the same job as the central part of Siesta (siesta.c) but includes some modification to allow interaction between the siesta loop and the Xforms events loop.

- **DyInteractionSim**: this function is called by DySiestaLike each time a new frame is read. In addition, it updates the current run and frame numbers displayed in the main panel. It gives the possibility to the user to interact with the GUI while frames are read, (for instance to use the Stop or Pause buttons).

- **DyStart**: called when the "Start" button of the main panel is pushed. It calls the functions *DyMakeSiestaFile* and *DySiestaLike*.

- **DySetPlot**: called each time you push a button like 1D, 2D, TIME, ... of the "Data and Plots" panel. It looks at the channels you selected and for each of them add a plot in the plots list (DyUpdatePlotBrowser) and sets its default parameters (DyPutDefaultInPlot).

- **DyEditPlot**: It does the same kind of job but this function is called when the "Edit" button of the plots browser is pushed. It opens the plot panel. Once you changed the plot's parameters, you push the button "Ok" and the function *DyEditPlotCallBack* is called. In this editing process, the following functions have been called: *DyEditPlot, DyPlotToPanel, DyRaisePanel, DyEditPlotCallBack, DyPanelToPlot, DyUpdatePlotBrowser, DyClosePanel*.

- **DyUpdatePlotBrowser**: called each time you add or delete a plot in the list.

- **DyPlotToPanel**: called each time you edit a plot. It puts values from the *Plot structure to the edition panel.

- **DyPanelToPlot**: called each time you have edited a plot. It puts values back from the edition panel to the *Plot structure.

- **DyPutDefaultInPlot**: called each time you create a new plot by pushing button 1D, 2D, TIME, ... in the "Data and Plots" panel.

- **UFrIFileCmIni, UFrIFileCmSim**: These are modified copies of the functions used in siesta to get frames from the network. Some added lines allow to interact with the data Display event loop. Are also available "UFrIFileOnlineSim" to read

frames online and "UFrIFileDdIni, UUFrIFileDdSim" to get frames from the Data Distribution server.

- **DyGetConfigfile**: called when you push the button "Load Config" of the Config File panel. This function calls DyGetConfig.

- **DyGetConfig**: This function reads a *.car file using the functions *DyGetClocks-FromConfig*, *DyGetFilesFromConfig* and *DyGetPlotFromConfig*. When you give to DyGetConfig a Siesta cards file, it only look at the siesta cards beginning with keywords UJMasterClocks, UFrIFile, UFrIFileOnline, UFrIFileCm, UFrIFileDd or UFrUnpack cards. Moreover, after a UFrUnpack card, there must be always a card beginning with UHplot1D, UHPlot2D, UHplotTime, UHplotFFT, UHpTfct or UH-sound.

- **DySetVar**: This function is called by the *DySetPlot* function or each time you select a channel in the "Data and Plots" panel. It takes the channel name and type (adc or sms data), used when a plot is created.

- **DyRaz**: It is called by the Clear buttons of some browsers and by the buttons "Cleaning" and "Reset All" of the main panel. Depending on the argument passed, it resets the input files list (arg=1), the plots (arg=2) (that means plots list, clocks list except the frame clock, and all the *Plot structure variables), the list of data to dump (arg=3), the dumped data panel (arg=4), the PAW graphical windows only (arg=5), or all the previously cited items plus the frame clock (arg=0).

3.5   Example of data display working steps

In the function main, *fl_do_loop()* is called within an infinite loop. This means that Xforms is indefinitely waiting for an X event (any change in the currently displayed panels). Once you got a configuration file (read via the *DyGetConfig* function) or once you got a frames source (whose first frame is read by *DyReadFirstFrame*), and once you set and edited your plots (*DySetPlot* and *DyEditPlot*), you can push the "Start" button (*DyStart* that calls *DySiestaLike*).

This action has a first consequence: the building of a configuration file "dd.car", This is a Siesta cards file, which is then read by the function *DySiestaLike*. Within the DySiestaLike loop, a periodic call to the function *DyInteractionSim* allows interactivity with the user. Each time you push the "Stop" button while DySiestaLike is running, a plots file "dd.hsto" is written to disk, and the DySiestaLike loop is stopped.

References

[1] *Frame format specifications: Internal notes LIGO-T970130-F-E and VIR-SPE-LAP-5400-102*
see also http://lapp.in2p3.fr/virgo/FrameL

[2] *Xforms*
    library (open source since 2001): http://savannah.nongnu.org/projects/xforms and
    http://savannah.nongnu.org/download/xforms

[3] *Siesta: see documentation in /virgoApp/siesta/xxx/doc*

[4] *Root library, for data handling and visualization: http://root.cern.ch*

[5] *Frame library: Virgo note VIR-MAN-LAP-5400-103, 1997*
    see also http://lapp.in2p3.fr/virgo/FrameL

[6] *CSet package: see documentation in /virgoApp/CSet/xxx/doc*

[7] *The Cm library: Virgo note VIR-MAN-LAL-5100-112, 1998*

[8] *Frame distribution through shared memory: Virgo note VIR-MAN-LAP-5400-113,*
    2000