

Mya Archive Data Viewer
Design Specification
August 29, 2007
By Chris Slominski and Michele Joyce

INTRODUCTION	2
OVERVIEW	2
LOOK AND FEEL	2
STORY BOARDS.....	2
<i>Time domain chart</i>	3
<i>Correlation charts</i>	5
<i>Live strip chart</i>	6
<i>Menu bars</i>	8
SIGNAL PROPERTIES	10
X AXIS PROPERTIES	11
Y AXIS PROPERTIES	13
PLOT PROPERTIES	13
MISCELLANEOUS DIALOGS	14
PREFERENCES	17
SIGNAL SELECTION	19
CONTEXT MENUS.....	20
THE ORDINATE AXIS ON MYA CHANNEL HISTORY CHARTS	21
<i>Adding to a chart, having one or more plotted channels</i>	22
COMMAND LINE INTERFACE.....	23
REQUIREMENTS ACCOUNTABILITY	23
DEPENDENCIES.....	25
TCL PACKAGES.....	25
TCL 'C' UTILITY PACKAGE	25
SWIG	26

Introduction

This document is a response to the Jefferson Lab Accelerator Software group's requirements for an archive data visualization tool¹. The document is offered as evidence of the designer's thorough understanding of the needs for this tool, and must be approved by software group leadership before a development effort will begin.

Overview

This graphical tool will operate on the hpux-11-11-parisc and rhel-4-0-ia32 computer systems used in the accelerator division. It will be written primarily in the Tcl/Tk programming language, with some components written as C++ shared object packages that are loadable into Tcl. The viewer tool will require the installation of the Tcl interpreter version 8.4 with threading, on both of the target computer systems.

Look and Feel

This section describes how the users will interact with the viewer tool and the appearance of the various graphical components they will be shown.

The user interface will be kept responsive to user and operating system events by utilizing the threading capabilities of Tcl 8.4. Operations that potentially may take as much as 200 milliseconds or more will be performed in an independent thread to keep the graphical interface fresh and responsive. Prototyping efforts have proven that large channel history fetches may be performed concurrently while maintaining a responsive interface using Tcl threads.

There are a number of default settings that govern the actions of the viewer tool. Each user will have a preferences file in their home directory, created by the tool. They will be able to edit this file to modify some behaviors of the tool. See the story boards below to see what preferences are provided.

One instance of the tool will not be usable with multiple Mya deployments. The sole Mya deployment is selected via the command line and is fixed at startup. If you want to use the tool to view channel history from another Mya deployment, start another instance.

Story boards

This section includes a number of *story board* diagrams that show the appearance of various graphical interface components. These are scanned hand drawn diagrams and notes from the interface designer.

The user will be presented three chart types; time domain charts, live strip charts, and correlation charts. The type of the initial window must be chosen on the program execution command line. Any number of additional chart windows may be opened, of any chart type, by use of the *File* menu provided on all chart windows. These chart types are shown in the following sections.

¹ http://jira/secure/attachment/10105/graphical_display_tool_srs_v1_2.pdf

Time domain chart

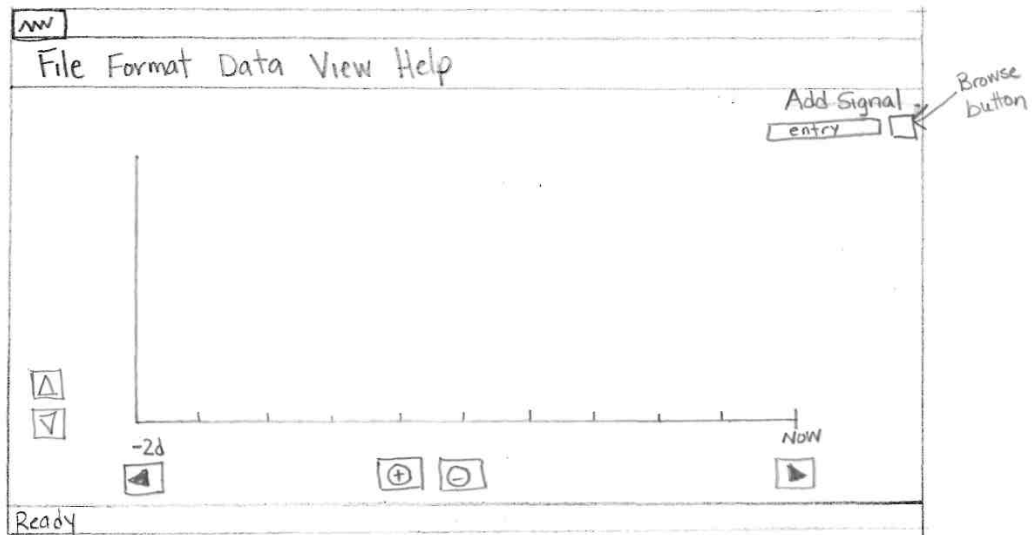
Figure 1: New Archive Plot prior to user having selected signals.

- The x axis will be configured with a default time range (user preferences).
- All arrow buttons will be disabled or hidden until appropriate.
- The user adds a signal (or signals) by either entering a signal name in the Add Signal “entry” field, or pushing the “Signal Browse” button for signal selection help and adding signals from there. Note that browsing only shows the names of signals that are being archived, which may not be useful when requesting live control system strip charts. Also note that though Tcl edit boxes support drag and drop, they are not compatible with the edm/medm channel name drop.

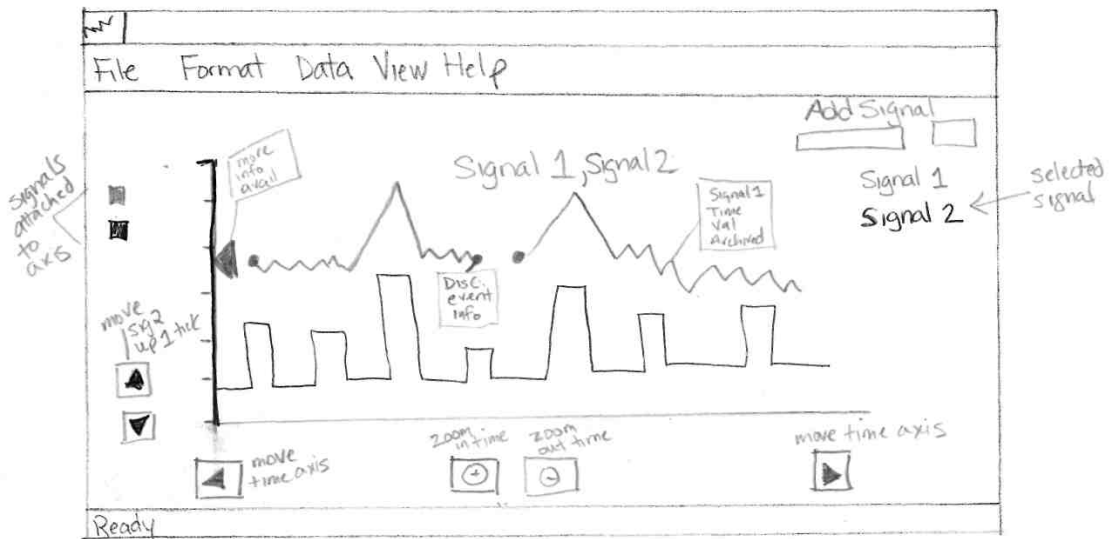
Figure 2: Archive plot with signals added.

- Time range (x axis) can quickly be slewed with the left and right arrows.
- “Zoom In” and “Zoom Out” buttons will respectively decrease and increase the time range viewed on the x axis. Zooms will work in steps of two tick marks.
- You may “*rubber band*” an area of the chart to perform a combined X and Y axis zoom.
- Y axis will correspond to the “selected signal”
- Other signals may also be attached to the same y axis. These signals will be indicated as icons to the left of the y axis.
- Selected signal may be “nudged” up or down the y axis to make viewing multiple signals easier.
- Disconnect events will be shown as distinct dots. Mousing over the disconnect event will provide the user information about the event.
- Mousing over a signal’s plot will provide the value and timestamp of the nearest data point, as long as the mouse is within close proximity of the point.
- Arrow indicators (in the signal’s corresponding color) on the edges of the plot will indicate that more data is available, if the axis is resized in the indicated direction.
- Axis stepping with arrow buttons will be performed in units of one minor tick mark.

New Archive Plot



Archive Plot w/ signals



Correlation charts

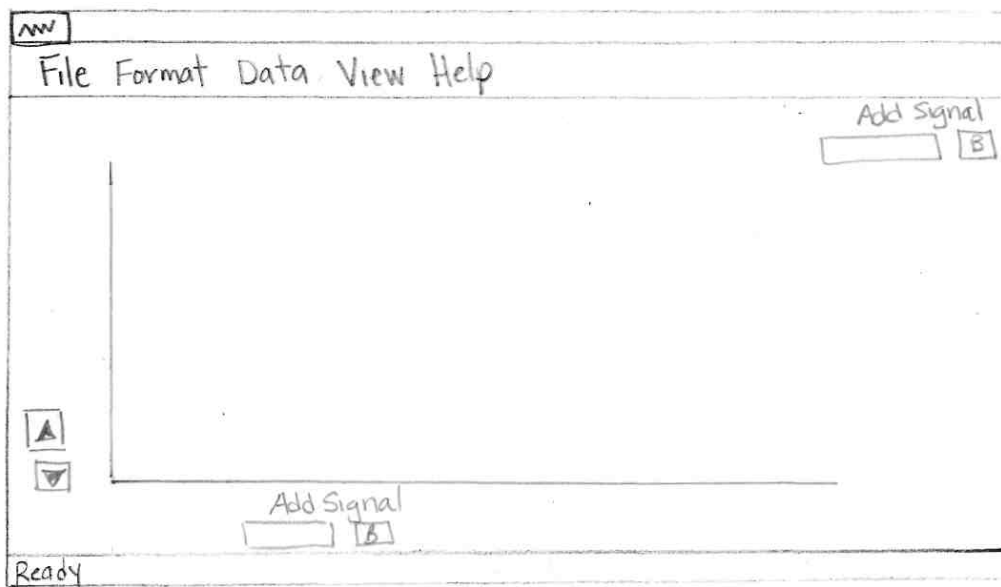
Figure 1: New Correlation Plot prior to user having selected signals.

- Use the “Add Signal” field on the x axis to add the independent variable to the graph. See the previous explanation of the legend’s “Add Signal” control.
- Dependant variables are added via the “Add Signal” field on the right side.
- You may change the independent signal, using the various X axis menus, and correlations to previously entered dependent channels will be recomputed.
- The default time range for a correlation may be modified via the *data->collection* menu.

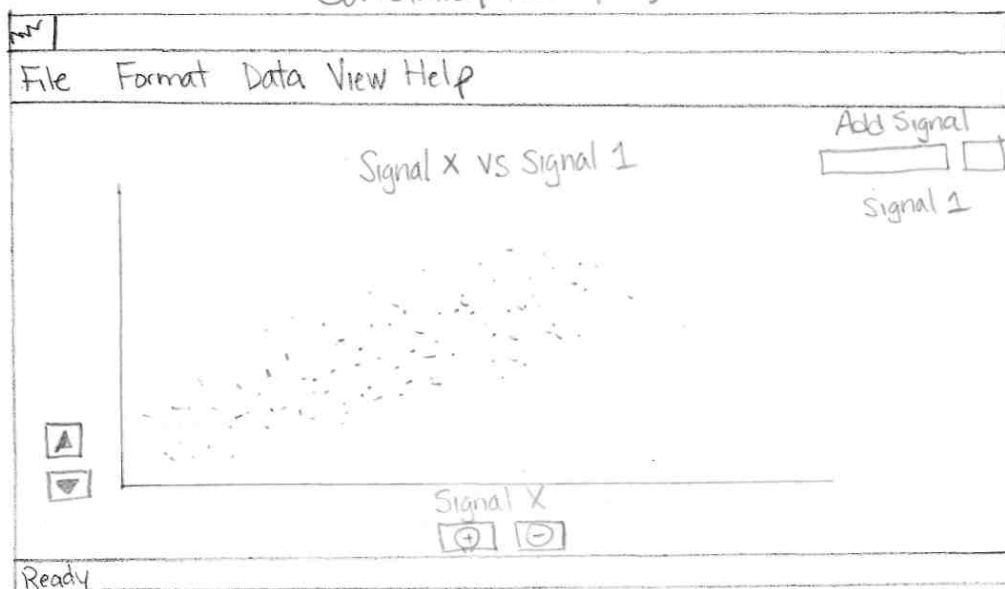
Figure 2: Correlation Plot with signals added.

- Correlation plot is shown without line segments.
- “Zoom In” and “Zoom Out” buttons will respectively decrease and increase the x axis range.
- Selected signal may be “nudged” up or down the y axis.

New Correlation Plot



Correlation Plot w/ Signals



Live strip chart

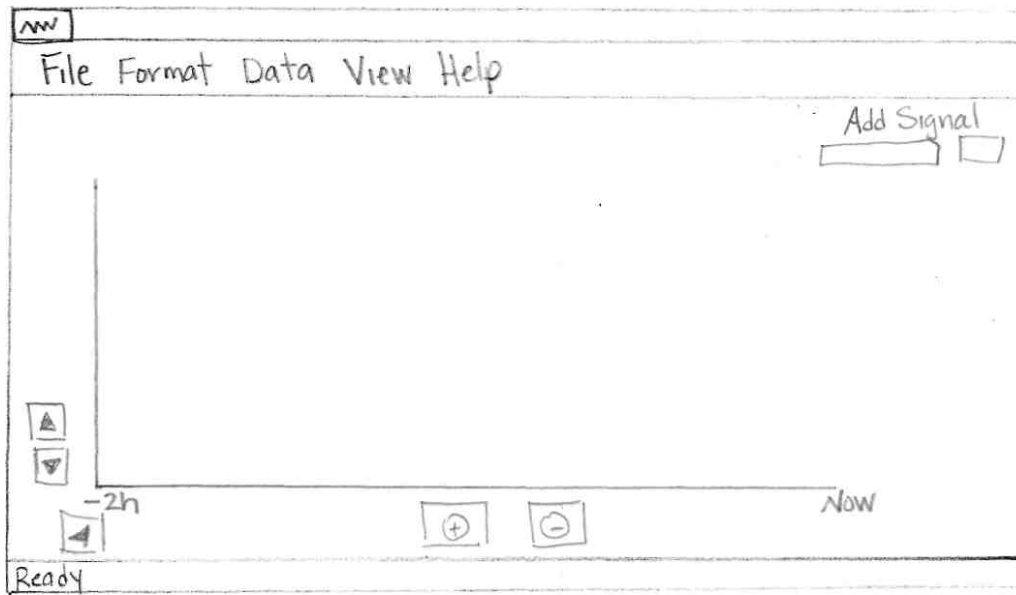
Figure 1: New Striptool Plot prior to user having selected signals.

- The x axis will be configured with a default “age” (i.e. 2 hours ago)
- Use the “Add Signal” field to add signals.

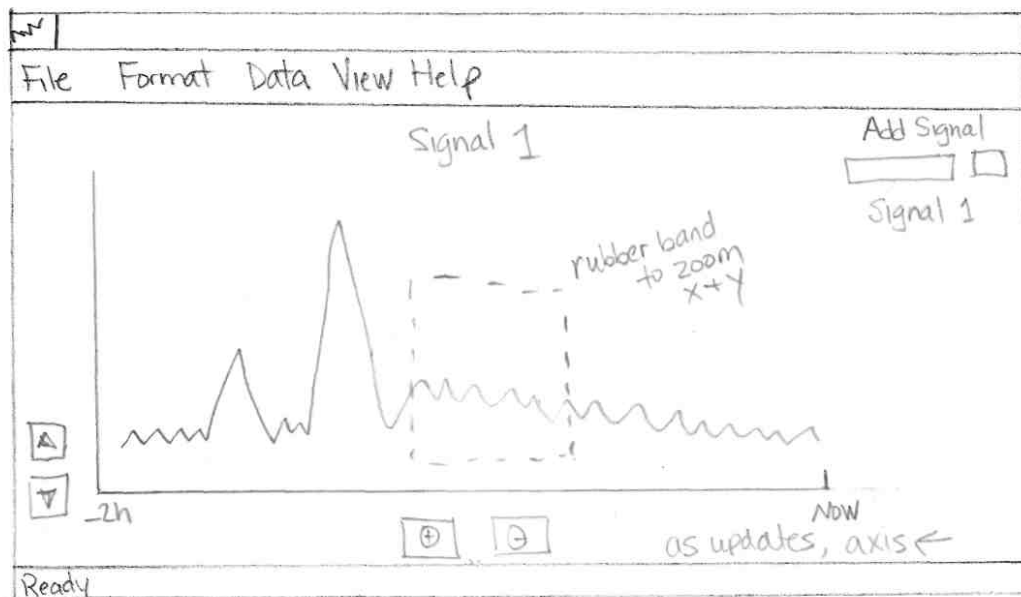
Figure 2: Striptool plot with signals added.

- As the striptool collects data, the x axis will automatically reconfigure so that the right most edge always corresponds to “now”
- “Zoom In” will decrease the represented time range relative to “now”. “Zoom Out” will increase the time range relative to “now”
- Selected signals may be “nudged” up or down the y axis”
- Rubber-band feature can be used to “Zoom In” x and y simultaneously. Right mouse click will “undo” the zoom.

New Striptool Plot



Striptool Plot w/ Signals



Menu bars

Menubar Menus

File

New... Archive Plot
Correlation Plot
Stripplot Plot

Load Config (loads predefined configuration file)

Close (closes this window)

Save Config (if multiple windows:

- Window1
- Window2
- Window3

) Check desired windows.

Save As

Print (prints this graph to selected printer)

Log (creates logentry w/ additional comments to specified log book)

Exit (close All open windows)

Format (change name to Properties)

Signals

x axis

y axis

plot

(displays properties for all signals, selected signal on top)
(displays xaxis properties)
(displays properties for all yaxis selected axis on top)
(displays plot properties)

Menubar Menus (cont)

Data

- Analysis... linear fit / hide fit
- Statistics (show stats for all signals)
- Collection (max, min points, source) [correlation + Archive only]
- Export (data, image dump)

View

- Signal Information (displays MYA info for all signals)
- Preferences (user default preferences)

Window 1

Window 2

Signal properties

If invoked from *Menubar=>Properties=>Signals* the resultant tabbed GUI will include properties for all signals (one per tab.) The selected signal will be displayed. If invoked from a specific signal's context menu, only the properties for that signal will be displayed.

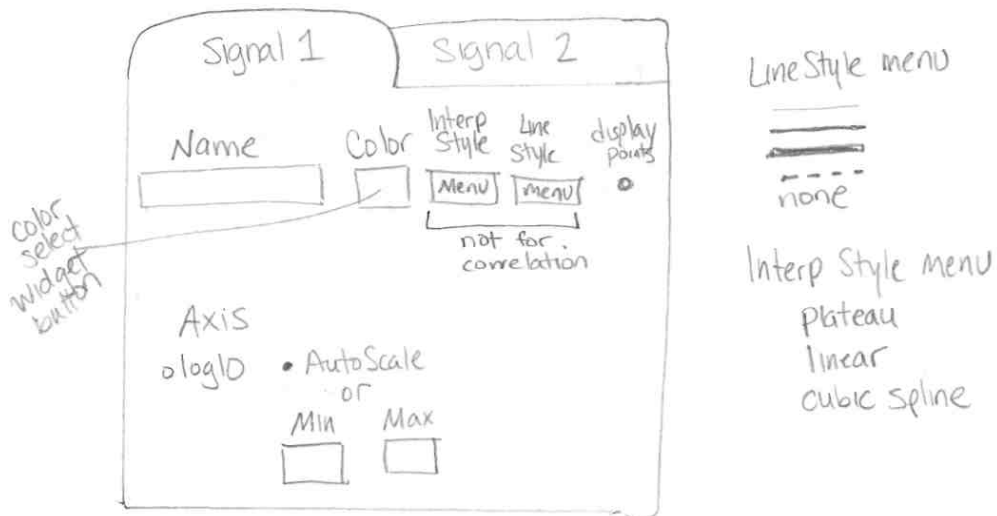
The following properties can be modified for an individual signal.

- label
- color
- interpolation style (plateau, linear, cubic spline, etc.)
- line style (thickness, dotted, none, etc)
- display individual points (on/off)
- adel (for striptool plot)

In addition, a signal's y axis can be managed.

- Log10 (on/off)
- Scaling (auto scale or max/min)

Signal Properties



Format → Signals
 show all signals in tabbed frame, selected signal on top
 Signal context menu → properties
 only show specified signal

X axis properties

This GUI can be invoked from either *Menubar* ⇒ *Properties* ⇒ *X Axis* or the X axis context menu. The GUI that is displayed is dependant on the type of plot.

Figure 1: Archive Plot

The following properties can be modified for an Archive Plot's x axis.

- Title
- Begin/End Time. The Begin/End times can be entered as absolute dates and times, relative, or a combination of both.

Figure 2: Correlation Plot

The following properties can be modified for a Correlation Plot's x axis.

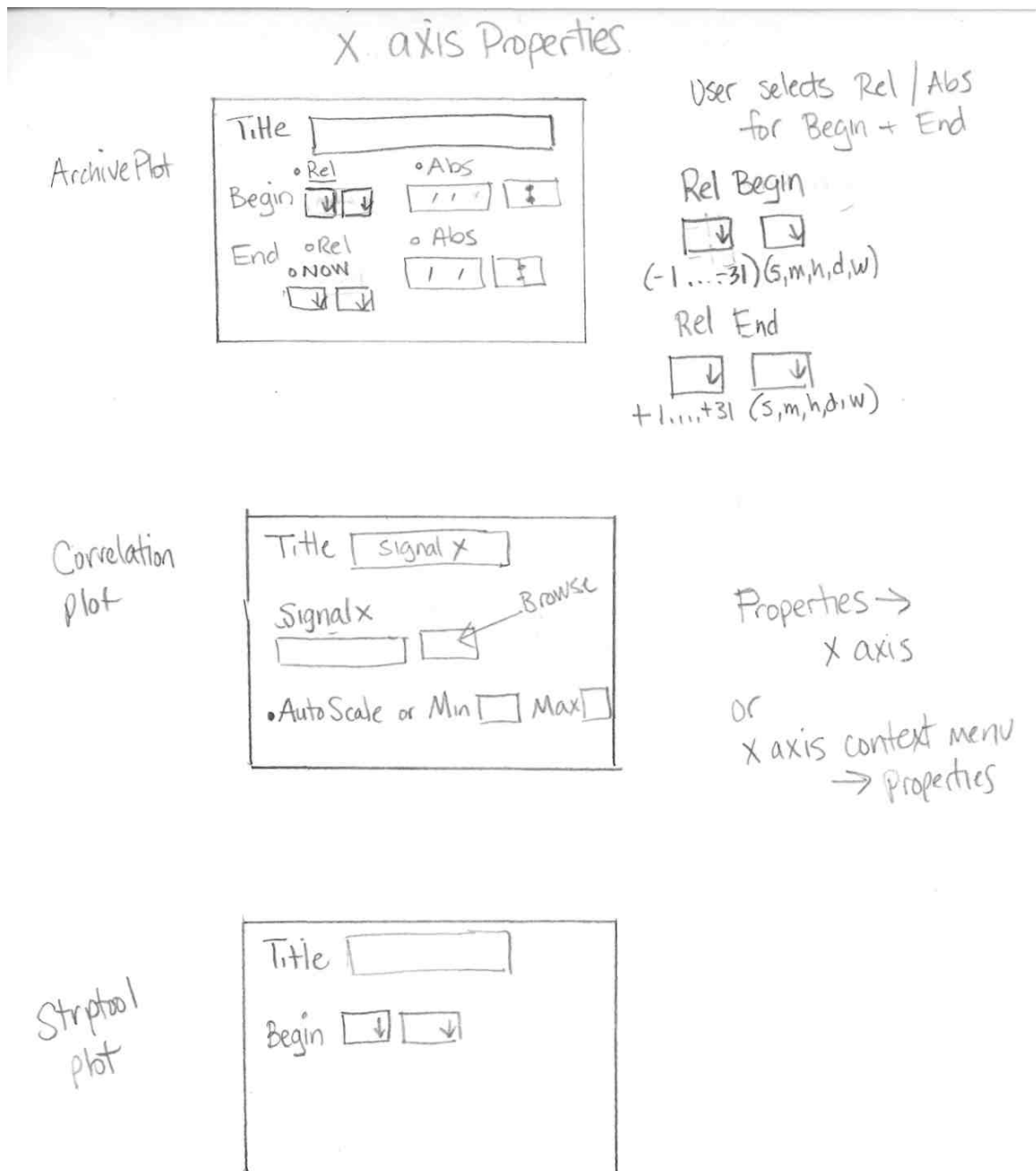
- Title (defaults to independent variable)
- Independent variable

- Scaling (autoscale or min/max)
- Begin and end times of data collection period.

Figure 3: Striptool Plot

The following properties can be modified for a Striptool Plot's x axis.

- Title
- Begin time (relative to "now")

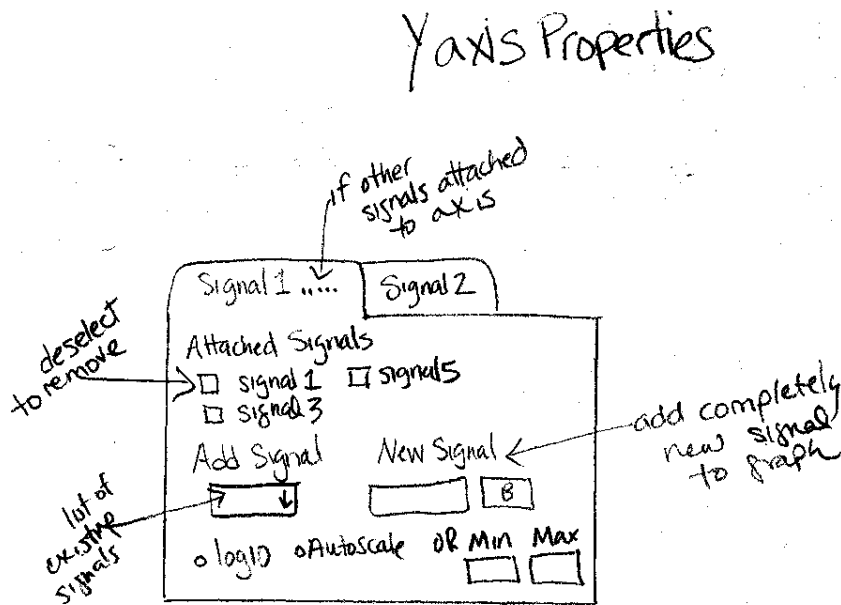


Y axis properties

If invoked from *Menubar=>Properties=>Y Axis* the resultant tabbed GUI will include properties for all y axes (one per tab.) The selected axis will be displayed. If invoked from the y axis context menu, only the properties for that axis will be displayed.

The following properties can be modified for a y axis.

- Attached signals (add or remove)
- Log scale enable (natural)
- Autoscale or min/max



Properties \Rightarrow Y axis

or

selected Y axis context menu
 \rightarrow properties

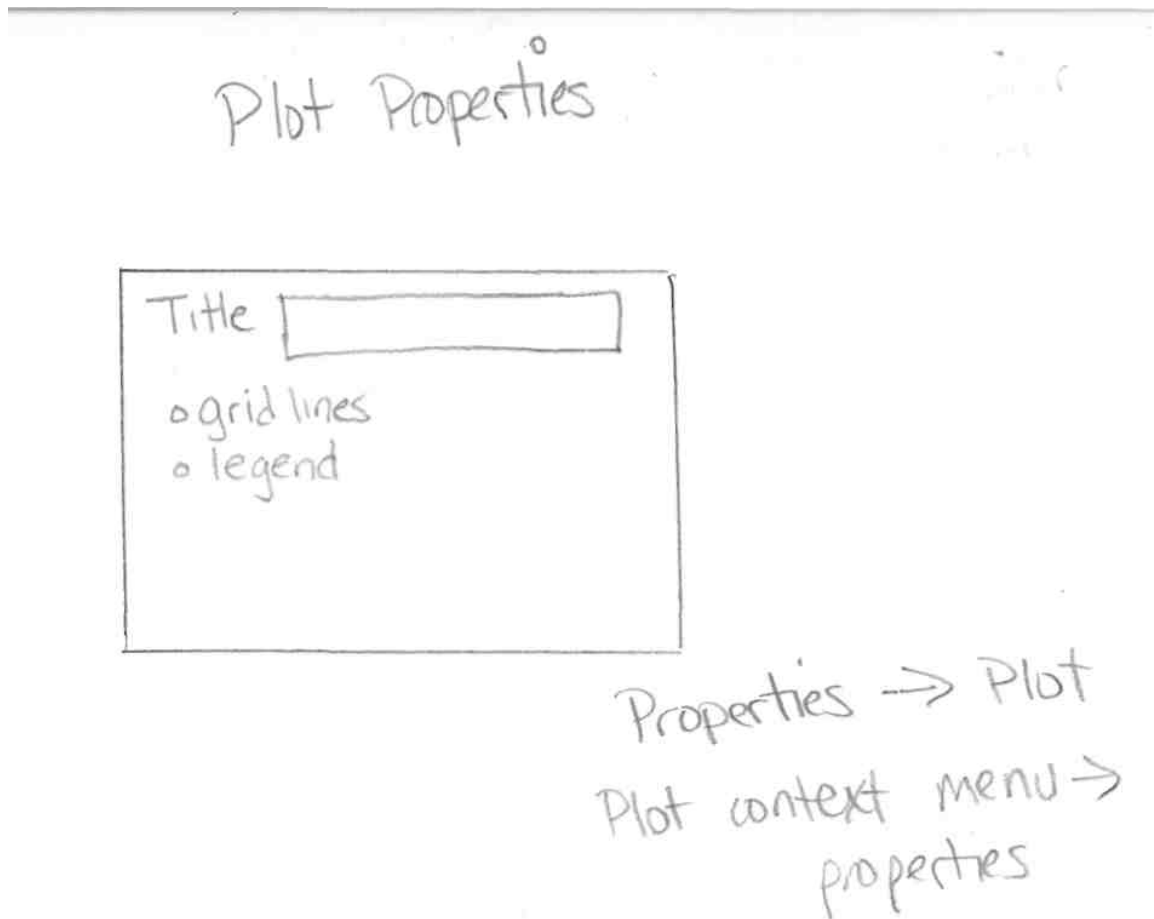
Plot properties

This GUI can be invoked from either *Menubar=>Properties=>Plot* or the Plot's context menu.

The following properties can be modified for the plot.

- Grid lines (on/off)

- Legend (display/hide)



Miscellaneous dialogs

Figure 1: Statistics GUI

If invoked from *Menubar=>Data=>Statistics* the resultant tabbed GUI will include statistics for all signals (one per tab). The selected signal will be displayed. If invoked from a specific signal's context menu, only the statistics for that signal will be displayed

Included on the statistics GUI (read only)

- EPICS channel name
- Min value
- Max value
- Mean value
- Standard deviation

Figure 2: Collection Information GUI

This GUI can be invoked from either *Menubar=>Data=>Collection* or the Plot's context menu. This GUI contains information regarding the collection of data from the archiver.

The following collection details can be modified from this GUI:

- Maximum number of points prior to bin request.
- Minimum number of points (for correlation plotting only)
- Time Range specification (for correlation plotting only)

Figure 3: Export GUI

This GUI can be invoked from either *Menubar=>Data=>Export* or the Plot's context menu. This GUI allows the user to export the plot image or the data to an external file.

Figure 4: Signal Information GUI

If invoked from *Menubar=>View=>Signal Information* the resultant tabbed GUI will include MYA information for all signals (one per tab.) The selected signal will be displayed. If invoked from a specific signal's context menu, only the MYA information for that signal will be displayed

Included on the signal information GUI (read only)

- EPICS channel name
- Archive Start Date/Time
- ADEL
- List of metadata keywords associated with this signal. User can then select metadata of interest.

Statistics Gui

Signal 1 PSET

Epics channel

min

max

mean

var

Data → Statistics

Signal context menu
→ properties

Collection Info GUI

Deployment

Max points

min points

Time range

Begin

End

Rel Abs

Rel Abs

Data → Collection
plot context → collection

min points +
range
only for correlation

Export GUI

Directory:

File name:

Format

• Data

• Image

Browse Button

User can export the
image Or the data
Menu selections TBD

Data → Export
Plot context menu →
Export

Signal Info GUI

Signal 1

EPICS Channel :

Archive Start :

ADEL :

↓

View → Signal Info
shows all signals

Signal context menu →
Info
only show selected

Preferences

The Preferences GUI will allow users to specify desired default behavior upon invocation. The preferences for all three types of plots can be found on the same GUI, and invoked from *Menubar* ⇒ *View* ⇒ *Preferences*

General Preferences (applicable to all plot types)

- MYA deployment
- Plot type (will allow user to invoke app from the command-line without specifying the plot type.
- Data export directory

Archive Plot Preferences

- Relative time range
- Maximum number of data points prior to bin request

Correlation Plot Preferences

- Relative time range
- Minimum number of data points
- Maximum number of data points prior to bin request

Striptool Plot Preferences

- Begin time (relative to “now”)
- Collection Adel (use ADEL from archiver, none, specify (?))

Preferences GUI

View → Preference

only relative selections

General

Deployment

Plot type

Data Dir

Archive Plot

Begin

End ☐ Now

max data points

Correlation Plot

Begin

End

☐ Now

min data points

max data points

Striptool

Begin

Collection Adel

☐ From archiver ☐ none

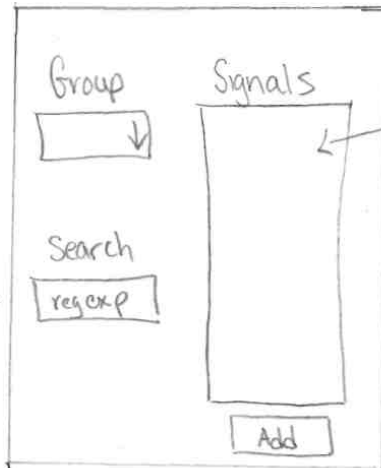
Signal selection

Wherever the user adds a signal to a plot, a Signal Browse button will be provided to aid in signal selection. The user populates the “Signals” list box by selecting a predefined group, or by entering a wildcard expression in an entry field. From the populated list box, the user can double-click on a single signal to add it to the associated plot, or select multiple signals and add them using the “Add” button provided.

Signal Selection Widget

Select a group,
and all signals
in the group
will appear in
signal list box

Search
for a
signal
w/ a reg. expres
and potential
choices appear in
signal listbox



<Double click> on signal to
add to plot

Select multiple signals +
push **Add** button
to add to plot

Add Signal

Context menus

Context Menus

Popup menu will be display when user right-mouse clicks on specified element

Plot

Hide/Show grid
Collection
Export

Properties

Title

Hide/Show
Edit

Legend

Hide/Show

Signal

Hide/Show
Remove
- Auto scale
- Statistics
Info
- properties

X axis

Hide/Show
Auto scale
- Properties

Y axis

Hide/Show
Auto scale
- properties

The ordinate axis on Mya channel history charts

This section states a set of rules that will apply to the ordinate axis shown on the viewer tool plots. This includes all three different chart types; static time domain plots, live data strip charts, and correlation plots. These rules dictate how a single ordinate axis will be shown when there may be multiple plots on one chart.

- A Channel history chart will have one or more ordinate axes. Only one axis will be visible at any time. Each axis will have a list of channels associated with it. Some of these channels may be active, while others are inactive. Inactive channels are not drawn on the screen, do not have any special color coded chart symbols shown on the chart, and do not impact auto scaling calculations of any axis.
- An axis may have its bounds explicitly set by the chart user. Alternately, the axis may be set to auto-scale, which will be a calculation of bounds made such that all active channels associated with the axis will take up the majority of the viewing height of the chart, centered as a set. The auto-scale bounds may change as the

bounds of the abscissa axis are modified, because the maximum and minimum values within the set of channels may change.

- Explicit ordinate axis bounds will remain in effect until the user requests new bound values or enables axis auto-scaling.
- Each channel plot coexisting on a single chart will have a unique color assigned to it. This color will also be used for the channel's legend entry and other unique chart symbols associated with the channel.
- The initial ordinate axis is established on a chart when the first channel plot is created. Auto scaling of the ordinate axis is used by default, but the user may later specify axis bounds via the menu system. The color of the created axis will match the color of the channel plot.
- When explicit axis scaling causes a channel plot to be entirely above or below the visible chart area, a color matched arrow will be shown on the appropriate chart edge indicating the direction to the plot.

Adding to a chart, having one or more plotted channels

- All channel plots added to a chart will be associated with the currently active ordinate axis. If the axis is in the auto-scale configuration, the scale bounds will be recalculated considering the impact of the newly added channel. Otherwise the plot will be made relative to the existing ordinate axis, even if the added plot will be entirely off screen.
- There will be a focus plot for a chart. It will be the last channel added to the chart; however the user can also select the focus plot by left clicking a channel name in the chart legend. Adding a new channel overrides a previously selected focus plot. The ordinate axis associated with the focus plot will be the only axis shown, with other axes hidden from view. The color of the ordinate axis will match the color of the focus plot.
- The color of the ordinate axis, including labels and tick marks, matches the color of the focus plot. Since an axis may have other plots associated with it as well, this association will be portrayed to the user by having a small color coded square for each associated non-focus plot. These squares will be stacked vertically just to the left of the vertical ordinate axis line.
- Right clicking on a channel name in the legend provides a pop up menu relevant to the channel that was clicked. One option will be to modify the channel's associated ordinate axis. Any changes made in this manner will only affect the axis of the selected channel. Therefore, if the channel belongs to a group of channels sharing an axis, the channel will be removed from the group and have an individual axis created for it. The user will be able to specify auto-scaling or explicit axis bounds. If the channel belongs to an auto-scaled set and the user wants to break it away and have its own auto-scaled axis, he will be able to do so.
- The menu system will provide the capability to modify the properties of the focus axis showing on the chart. Changes to the focus axis affect all plots that are associated with the axis. An exception to this is changes made to the focus axis by stepping the focus plot up or down using the ordinate axis arrow icons. When this

is done, the focus plot will be automatically removed from any channel group sharing the axis, so that it can be separated individually from other plotted channels. The user will also be able to add or remove plots to the focus axis via the menu system. When a plot is removed from the focus axis, it will be given its own unique axis set to auto-scale.

Command line interface

The program will have a number of command line options that alter startup behavior. These are itemized below.

- -mya; Name of the Mya deployment (defaults to 'ops')
- -type; One of *Archive*, *Correlation*, *Striptool*
- -config; Specify a saved chart(s) configuration
- -signal; Specify signal name(s)
- -xsignal; Specify independent correlation signal
- -begin; Begin time (archive or striptool)
- -end; End time (archive only)
- -print; Create chart and auto print to elog, then exit.

Requirements Accountability

This section addresses the sections from the viewer tool requirements that are not obviously met by this design, needing some form of discussion or disclaimer.

3.1.1-3 *It must be clear to the user whether the data displayed is archive data or live data. Data must be presented in a clear way that doesn't require the user to look at the time axis and see that the data is for the current time.*

We will provide this by having a vertical line on the strip chart, separating the archived data from the data collected live. It will move to the left until it goes off screen. When there is not separator line, it means the whole chart is filled with data that was collected live. There will only be one line, even when more than one channel is being plotted on the chart. The design of the strip chart is such that the chart "NOW" position is always reset each time there are any additions to the chart or modifications to the time axis.

3.1.3-1 *The user will be given the ability to specify the begin and end time of the required data for each channel specified in the history data request.*

The design assumes that each channel plotted on a single time domain chart shares a common time (x) axis. If you want to see channels at different time, open another time domain chart.

3.1.9 *The user should be able to specify which channels to plot from the command line, their time ranges, plot window screen location, and window size such that when executed the tool loads the needed data and plots them based on the command line switches. The user must at least specify the channel name and a reasonable default will be used for all other parameters that are not specified.*

The user should also be able to specify a file that contains all the information as well. Additionally a user will be able to save the current plot configuration to a file that can be

opened in the future and will recreate the current plot window locations, window sizes, channels, etc.

The design does not provide command line window positioning and size specification, nor does it save these to its configuration file. There is a variety of desktops in our environment, including standard single screens, triple wide screens, and the MCC Wall. We intend to allow the window managers to allocate desktop space, as they do a good job.

3.1.10-7 If there is a known bar of uncertainty for a channel the tool will be able to graphically display the uncertainty range.

The requirement was deferred after requirements review.

3.1.10-9 The user needs to be informed if the data points that are rendered are recorded data points or are interpolated points.

The design will allow the option of having actual data points in a curve indicated by symbols that will stand out from the lines that connect them.

3.1.10-17 The tool will allow the user to place one or more cursors on the plot for each channel. Cursor in this case is used to mean a small graphical element indicating an x,y coordinate on a plot line. The user will be able to “dial” the cursor left and right like the cursor function of an oscilloscope. The tool will display the x,y values at each cursor location. The user will have the ability to add two cursors to a plot line and ask the tool to compute the difference between the two cursor locations i.e. $x_2 - x_1$ and $y_2 - y_1$.

The requirement was deferred after requirements review.

3.1.11-1 The graphical plotting must allow for multiple independently configurable Y-axes scales. The user should be able to select the Y-axis scaling per channel when creating a plot. The user will be able to specify a minimum and maximum value of the axis, the number of ticks on the axis, and the scaling type i.e. linear and logarithmic.

3.1.11-2 The tool will provide a configurable linear time axis. The user will be able to specify a minimum and maximum value of the axis and the number of ticks on the axis.

The design does not provide for user specified tick mark formatting. This would add significant complexity to the product and provides minimal benefit to the user. The graphing package build in axis formatting does a very good job.

3.1.12 Data analysis

The requirement was deferred after requirements review.

3.1.13 Annotations

The requirement was deferred after requirements review.

3.1.14 Data selection

The requirement was deferred after requirements review.

3.1.16 Bit masking

The requirement was deferred after requirements review.

3.4.2 The tool will be able to process at least 4000 total data points per second i.e. the sum total of all data points for all requested channels.

Benchmarking indicated we were actually able to achieve 160,000 data points per second, which does not include plot time, however this has been observed to be quite small.

Dependencies

The archive data viewer will depend on a number of external software items. These dependencies are discussed in this section.

Tcl Packages

These are packages available for Tcl that will be used in the archive data viewer Tcl code.

- BLT – Graphing widget
- Tix - Widgets
- Thread – Threaded Tcl applications.
- Ttrace - Allow threads to share procedures .
- BWidget - Widgets
- logXml – In house package for the electronic logbook interface. It is currently not certified, but can easily be put into configuration control.

Tcl ‘C’ Utility Package

There will be a portion of the viewer tool code written as a C++ shared object, which will be loaded as a Tcl package. This will be done for efficiency and smooth interface to the certified C++ MyAPI and jlablib libraries, which provide channel history to Mya clients and a wrapper of EPICS channel access. The following is a list of features this package will provide. Refer to the *MyaViewer C++ Tcl Packages* document for a detailed description of this API.

- Fetch the number of channel events that occurred within a time span.
- Fetch channel events that occurred within a time span. This information will automatically be segregated into disjoint channel history segments, having the reasons for gaps in continuity provided as text messages.
- Fetch metadata keywords for a channel.
- Fetch metadata for a channel, identified by keyword.
- Perform a channel correlation with an independent channel.
- Get a list of Mya deployments.
- Set a particular deployment as active.
- Get archive group names.
- Get the names of channels within a group.
- Get the names of channels that match a regular expression.
- Get the start date of a channel.
- Provide a live epics channel monitoring thread for the strip chart feature. This will insert channel update events into the main Tcl thread’s event queue.

SWIG

The swig is an automated tool for creating Tcl wrappers for C/C++ libraries. It will not be used for the MyAPI and jlablib libraries because of the complexity of the interface and the custom nature in which they will be used. It will however be used for the certified cpplib library to give access to the time handling functions and the *Options* class.