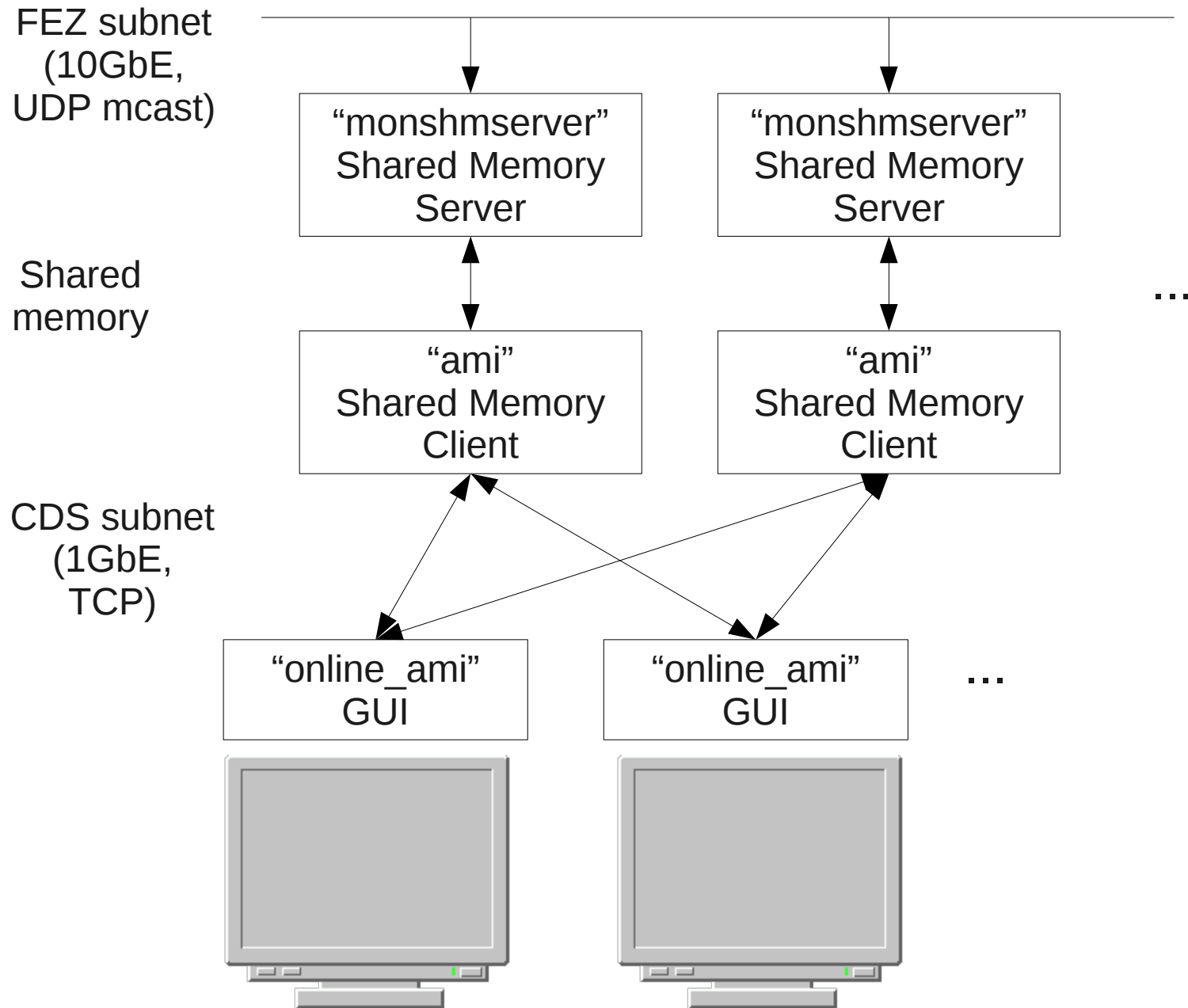


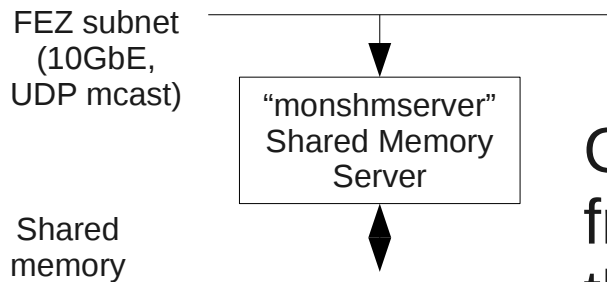
Online Analysis Design

Matt Weaver
January 17, 2012

Apps



monshmsserver



Generic application to receive datagrams from the DAQ private network and push them into shared memory.

Analysis designs are independent of DAQ network/partitioning design

Transitions are cached for late-arriving clients, so clients can learn the full current DAQ state.

L1Accepts are served promptly or dropped.

monshmsserver - parameters

`monshmsserver -p <platform> -i <node mask> -n <buffers> -s <buffersize> -P <shm tag> -u <tag ext> -c <nclient queues>`

platform distinguishes separate DAQ instances on the same network

node mask is a bit mask of recording nodes' data to monitor

nbuffers is the number of event buffers allocated in shared memory

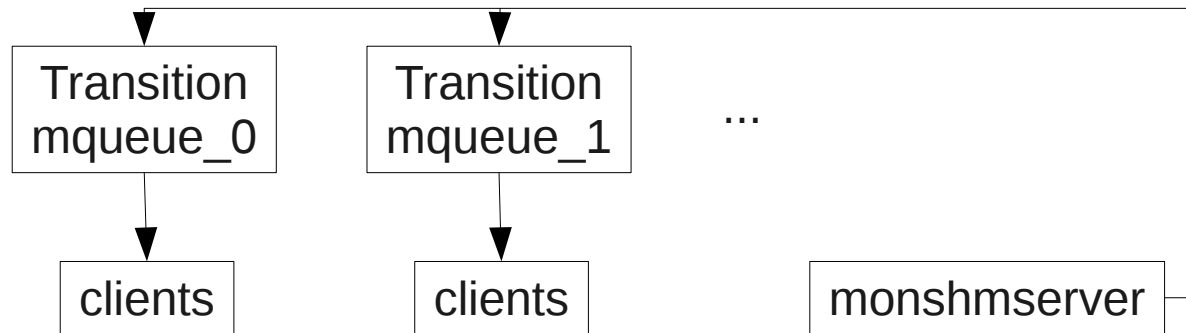
buffersize is the size of each event buffer in bytes

shm tag is a string used to identify the shared memory segment and the message queues

tag ext is merged with *<shm tag>*

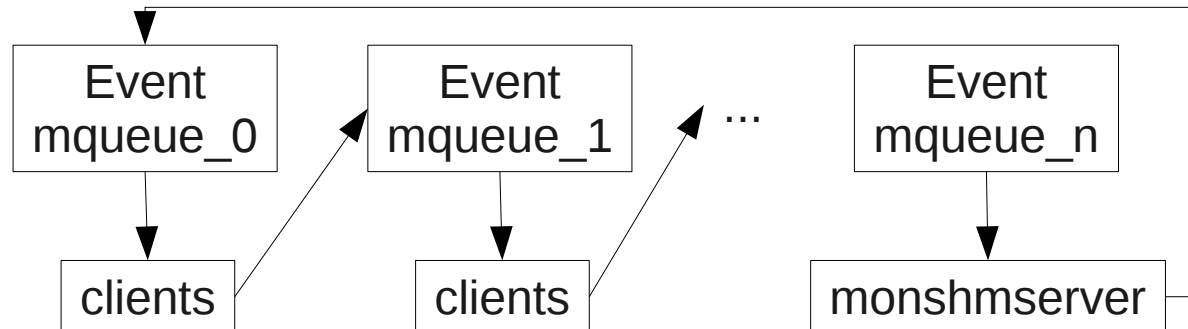
nclient queues is the number of serial message queues to setup

monshmsserver - dataflow



Arriving transitions are bcast to all queues.

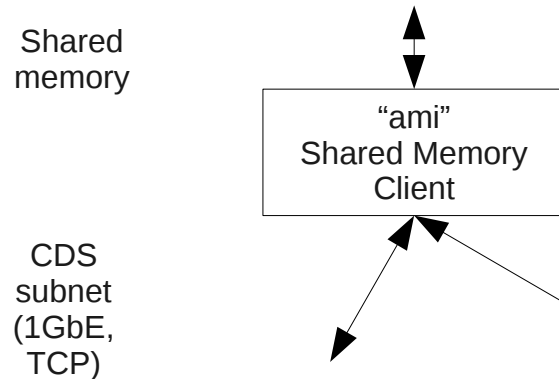
New clients are served cached transitions (ucast) to reach the current DAQ state.



If a free event buffer is available, arriving L1Accepts are ucast to the first queue.

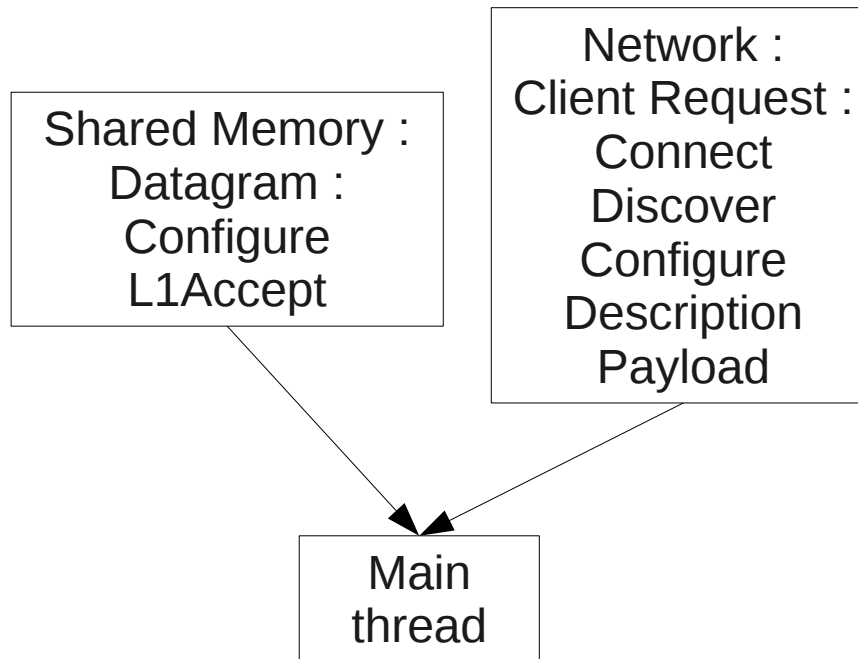
ami

Custom application to receive datagrams from shared memory, perform analyses, and export viewable data (plots).



Listens for TCP connections from analysis clients. Clients make requests to the *ami* server for analyses and updated plots.

ami - inputs



One shared memory input

Many network clients

Processing of client requests is serialized with datagram handling.

ami – handling client requests

Discover – transmit description of input event data set
list of scalars {EPICS vars, diodes, ...},
waveforms, images

Configure – setup all analyses associated with the client
Analysis : input data, event filter, operator, output data
input data : from the discovered data or
from the output of another Analysis
event filter : none, var range, logic and, logic or
operator : none, average, integral, entry math, bin math,
edge finder, peak finder, xy projection, $r\phi$ projection,
history chart, ...

Description – transmit description of output data set
list of plots

Payload – transmit output data set

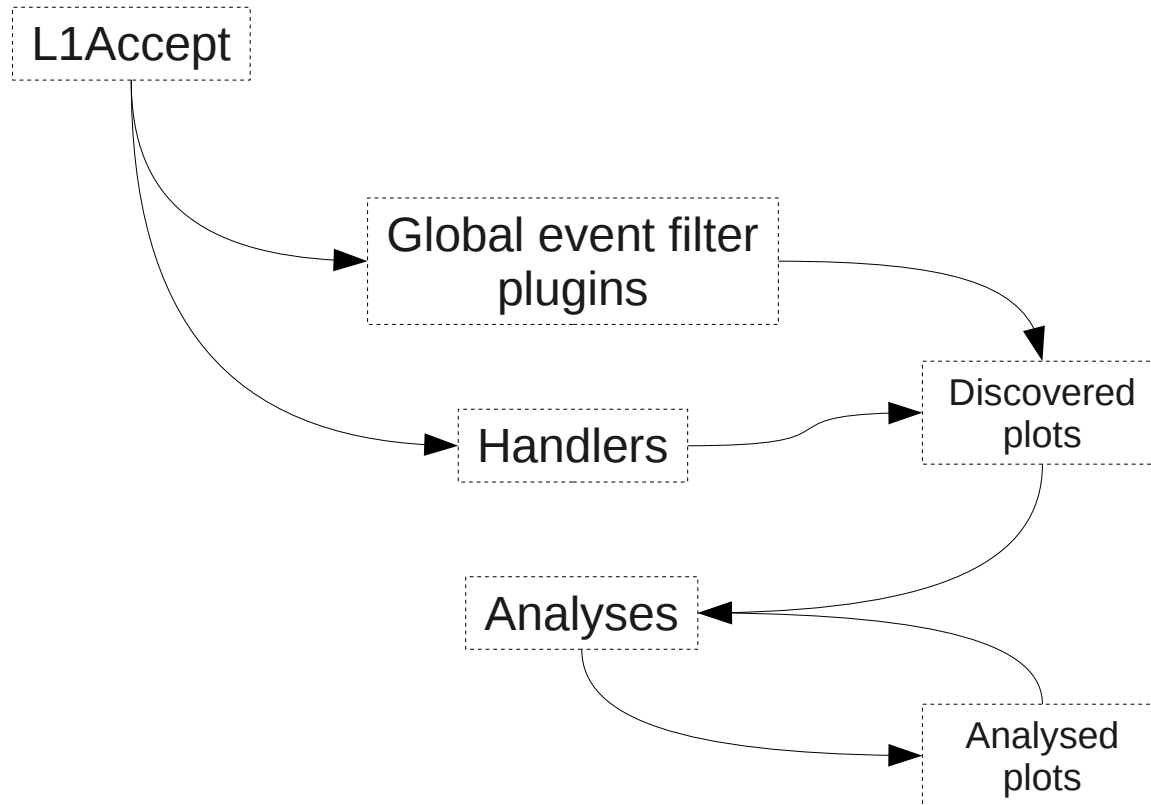
ami – handling *Configure* datagram

Destroy all previous handler/analysis setup

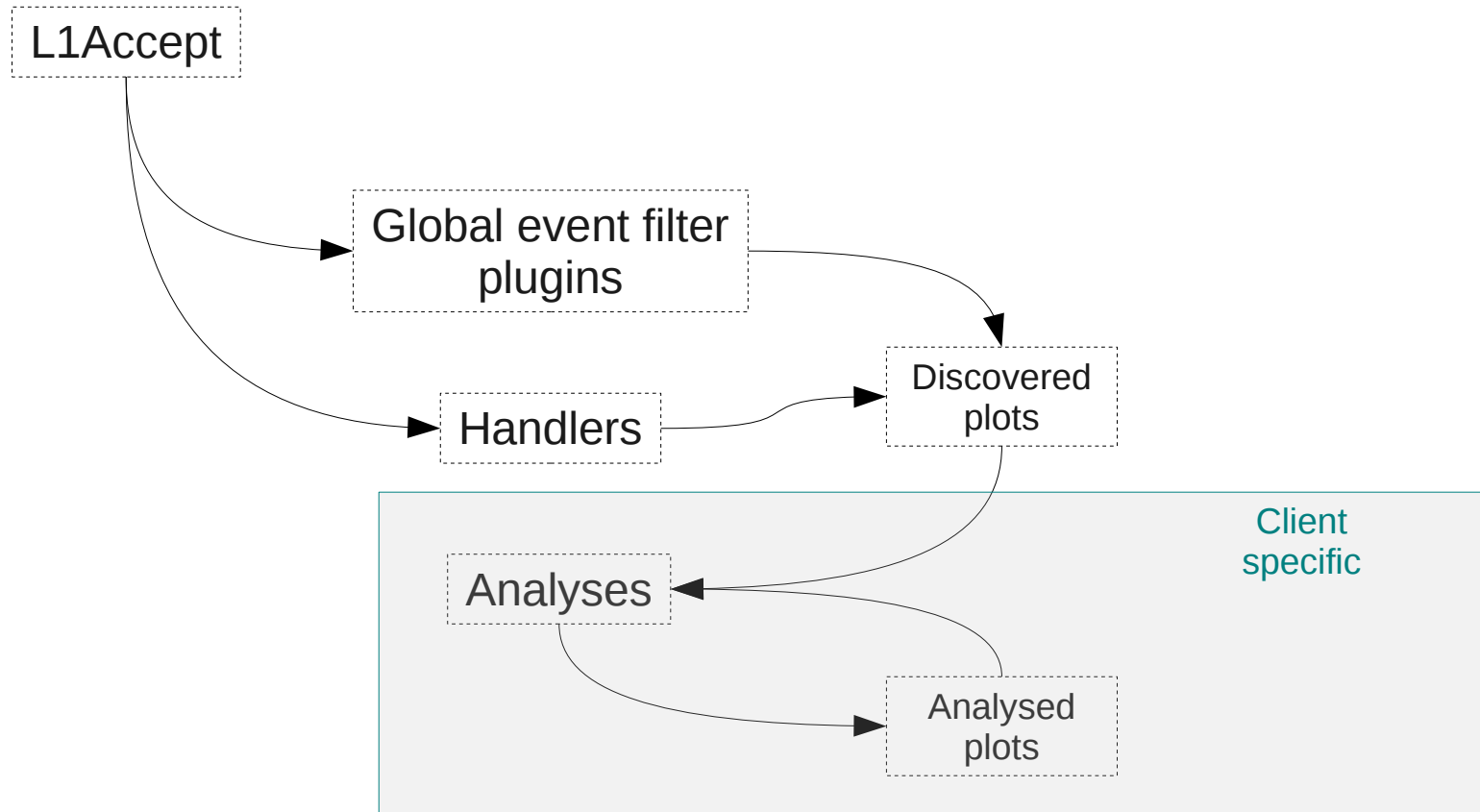
Discover data stream components {detectors,BLD}
and register handlers

Advertise discovered components (and plug-in modules) to
network clients.

ami – handling *L1Accept* datagram



ami – handling *L1Accept* datagram



Order matters:

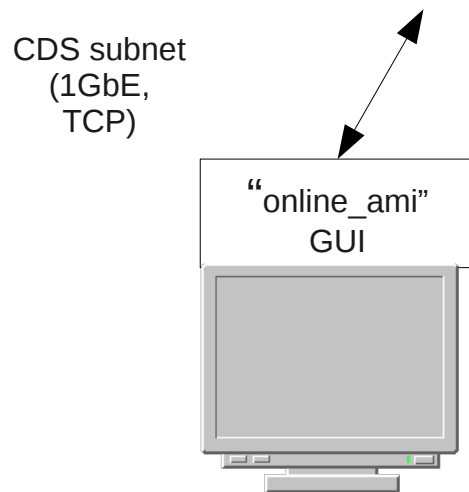
Each Analysis' input data must be computed beforehand.

It is up to the clients to configure the analyses in the correct dependency order.

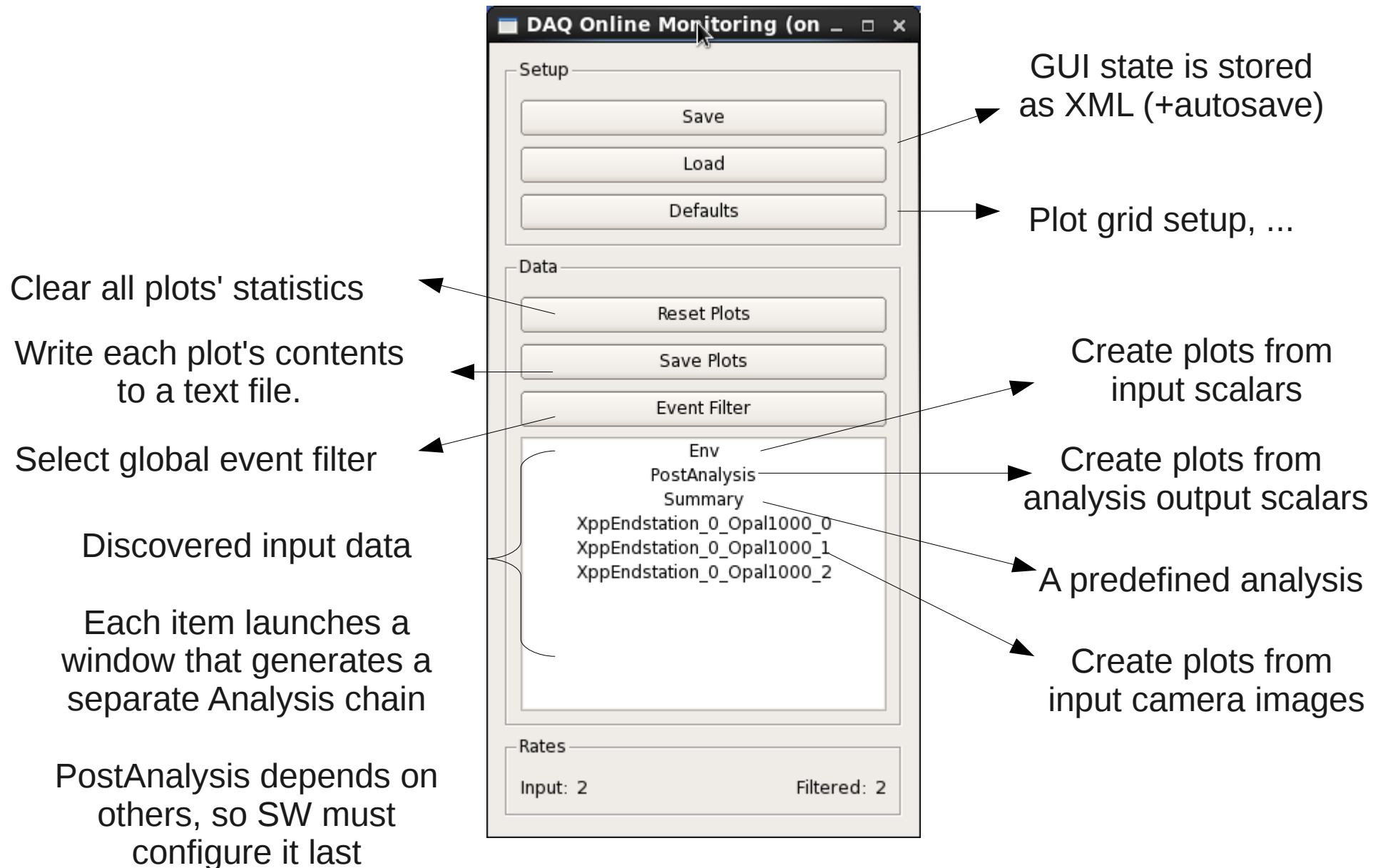
online_ami

Qt-based GUI that serves as a network client to ami.

Receives users' analysis configurations and displays resulting plots.



online_ami – main window



online_ami – “Env”(“PostAnalysis”) display

Update displays once or continuously

Type variable name or Select/Calculate

Define event filter

Add variable to PostAnalysis as *title*

Generate 1d histogram with specified binning

Generate history chart with prescaled time

Binned 2d plot (versus Calculation)

Scatter plot (versus other variable)

Define a variable to divide into all others
(flux normalization)

Weight averages by another variable

$$\frac{\sum x}{N} \rightarrow \frac{\sum w * x}{\sum w}$$

Env (on daq.xpp-ana01)

Run Single Rate(Hz) 2.5 Processed

Source Channel

BLD:EBEAM:L3E Select Filter

Plot Type

Title title Post

☐ Sum (1dH) bins 40 lo 6340 hi 6390

☐ Mean v Time points 100 prescale 1

☐ Mean v Var EE:GDET1:PMT1:ENRC X Var

bins 100 lo 0 hi 0.25

☒ Mean v Scan BLD:FEE:GDET1:PMT1:ENRC pts 200

Normalization

Normalize ☐ X ☐ Y variable to

Weighted Average

☐ Weight by

Plot Close

online_ami – “Env” display

Update displays once or continuously

Type variable name or Select/Calculate

Define event filter

Add v

Generate

Generate

Binn

Scat

Define

W

Plot interface
reused everywhere
a scalar variable is
produced

$$N \quad \sum w$$

binning

time

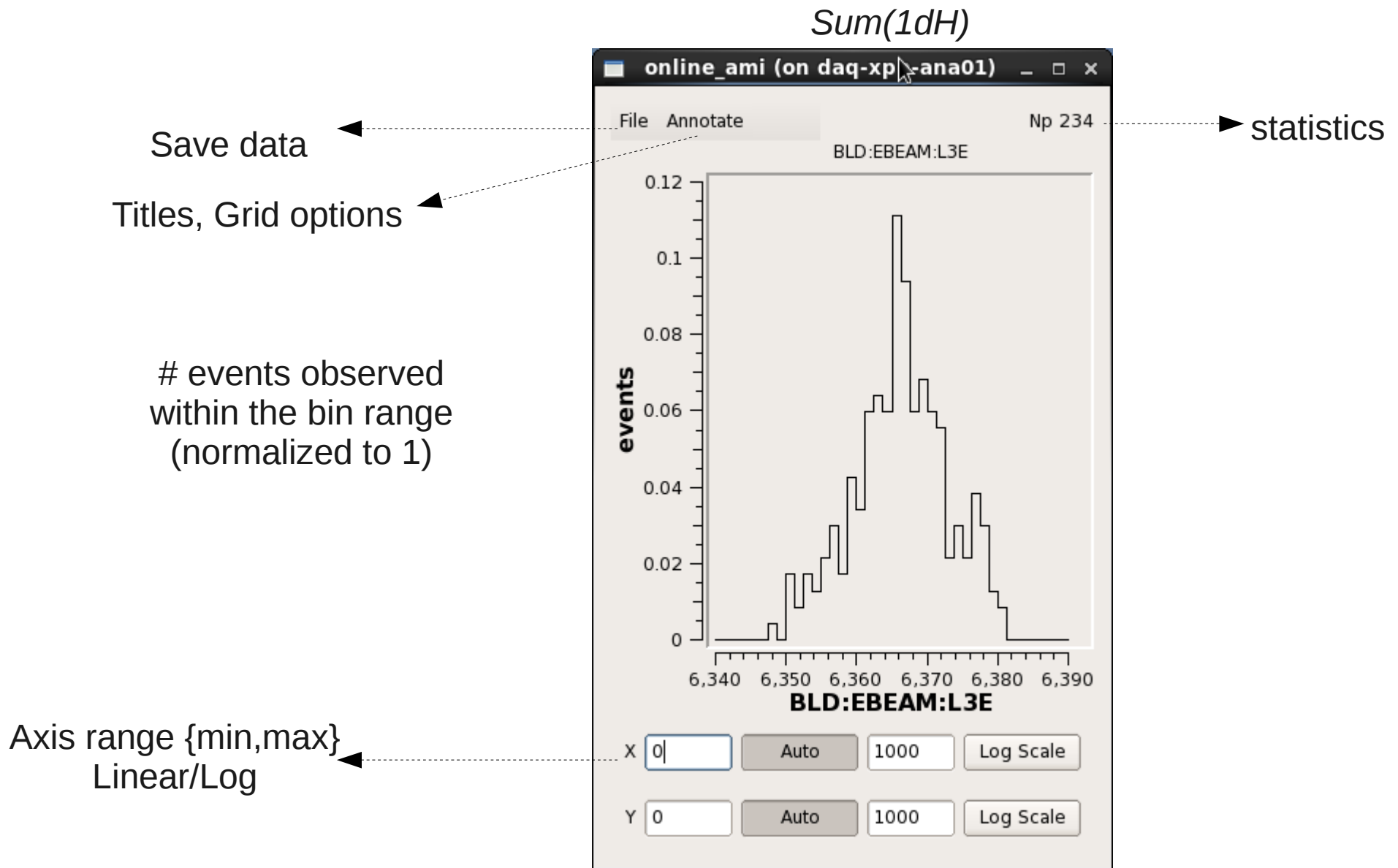
thers

riable

The screenshot shows the 'Env (on daq.xpp-ana01)' window with the following settings:

- Run/Single:** Buttons for 'Run' and 'Single'.
- Rate(Hz):** A text box containing '2.5'.
- Processed:** A status indicator.
- Source Channel:** A text box containing 'BLD:EBEAM:L3E' with a 'Select' button and a 'Filter' button.
- Plot Type:**
 - Title:** A text box containing 'title' and a 'Post' button.
 - Sum (1dH):** A radio button option with 'bins' set to '40', 'lo' set to '6340', and 'hi' set to '6390'.
 - Mean:** A radio button option with a 'v Time points' set to '100' and 'prescale' set to '1'.
 - Mean v Var:** A radio button option with 'EE:GDET1:PMT1:ENRC' in the 'X Var' text box, 'bins' set to '100', 'lo' set to '0', and 'hi' set to '0.25'.
 - Mean v Scan:** A selected radio button option with 'BLD:FEE:GDET1:PMT1:ENRC' in the 'X Var' text box and 'pts' set to '200'.
- Normalization:**
 - Normalize:** A checkbox that is unchecked.
 - variable to:** A text box.
- Weighted Average:**
 - Weight by:** A checkbox that is unchecked.
- Buttons:** 'Plot' and 'Close' buttons at the bottom.

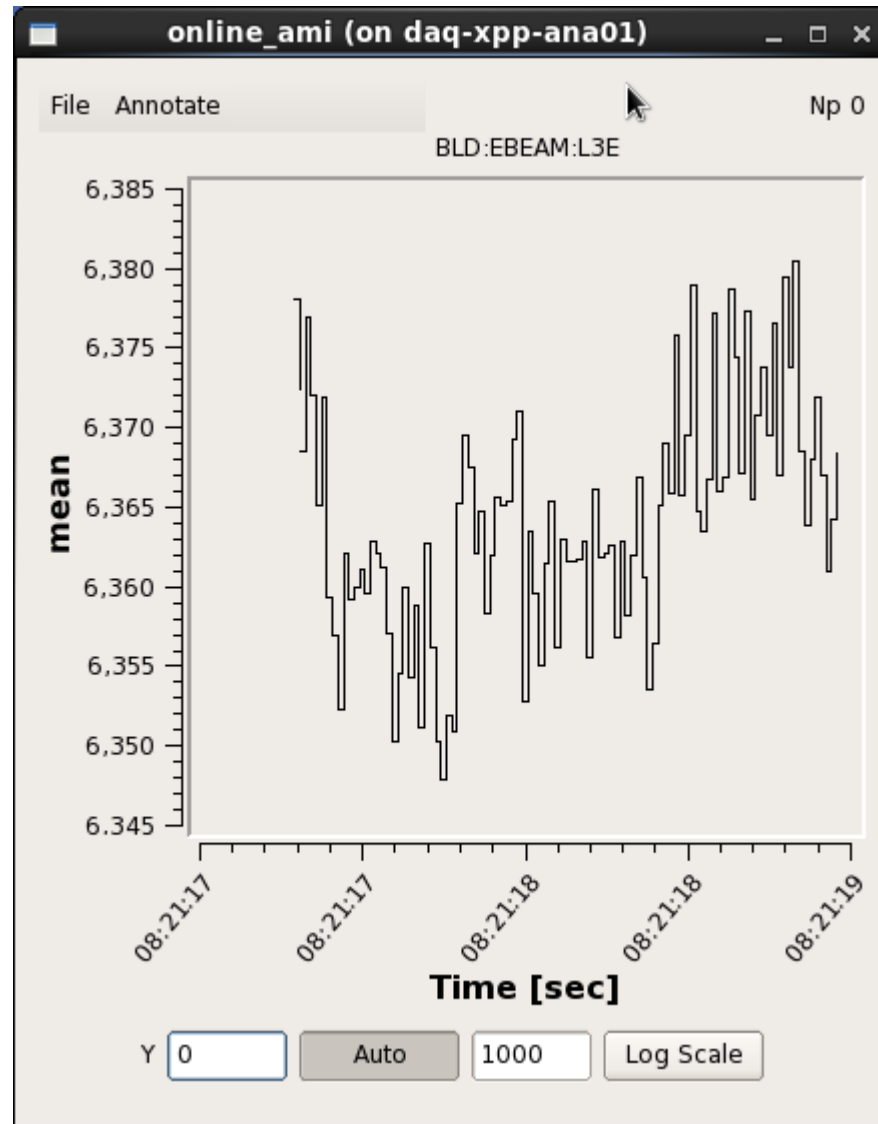
online_ami – “Env” display



online_ami – “Env” display

Mean v Time

Each point is the
mean of *variable*
in events observed
between plot updates

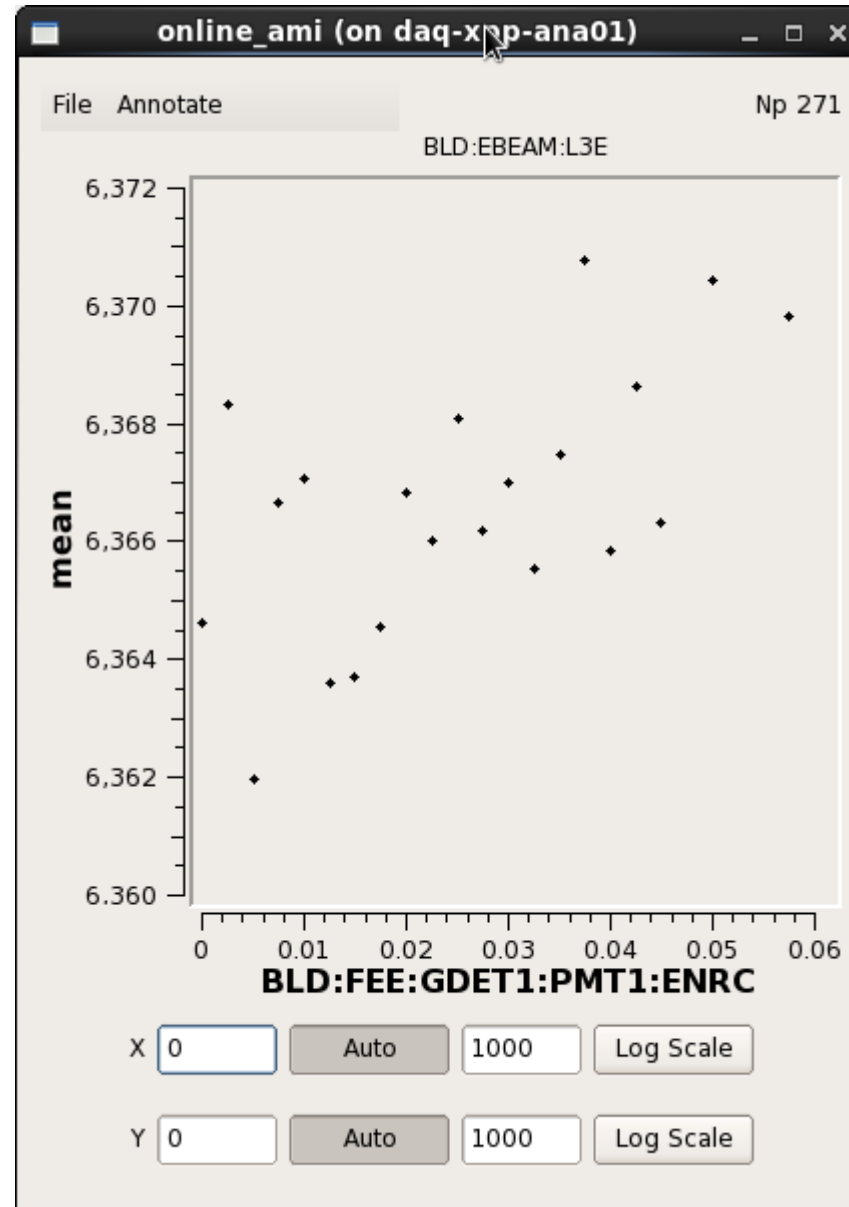


Time from most
recent event's
timestamp

online_ami – “Env” display

Mean v Var

Y-axis is mean of
variable for all
events where *x-var*
is within bin ranges

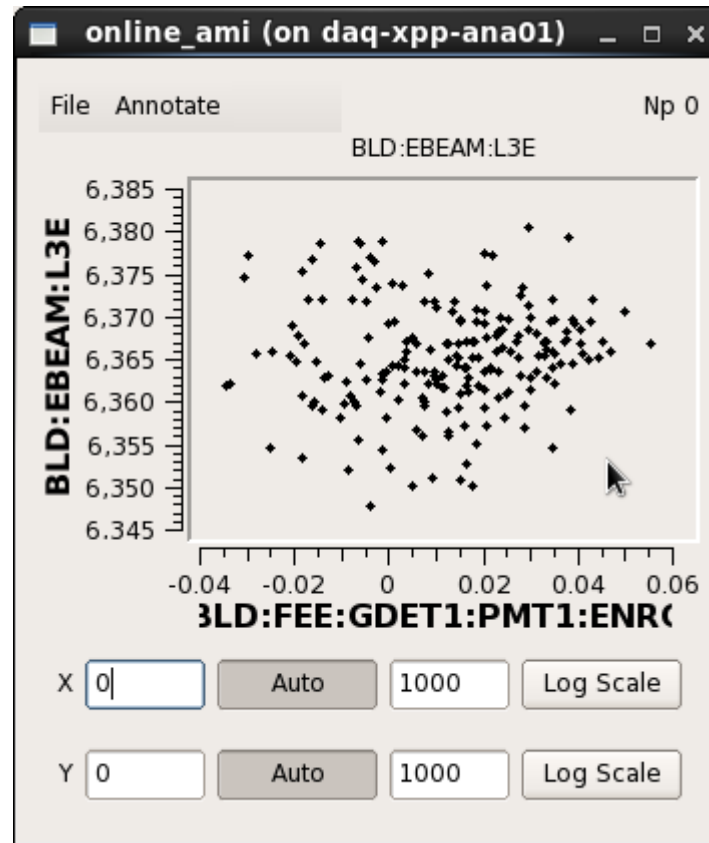


X-axis is equidistant
bins

online_ami – “Env” display

Mean v Scan

Y-axis is mean of
variable for all
events where *x-var*
is the same

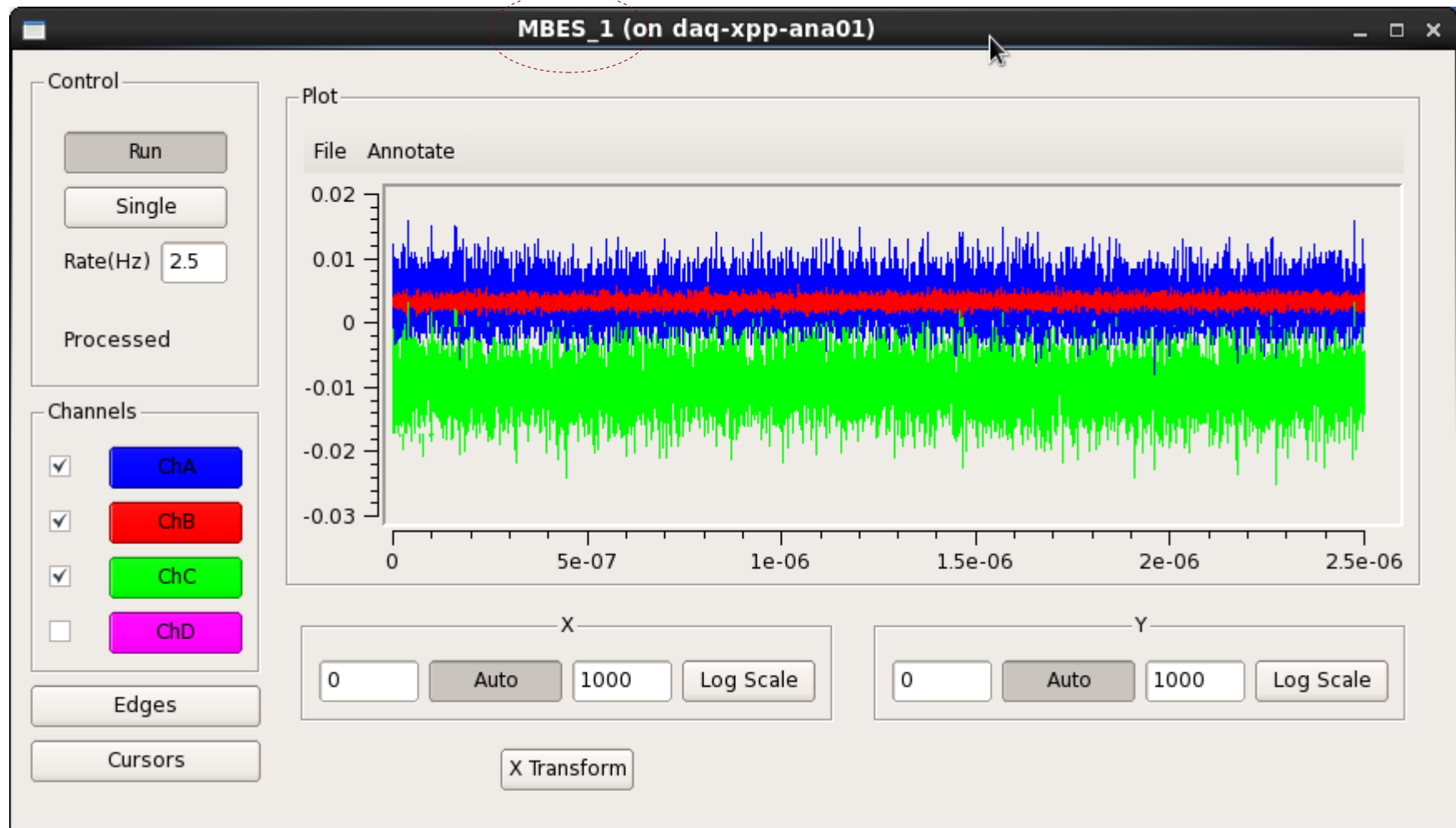


X-axis is most recent
N unique values

online_ami – waveform display

MBES - generate analysis derived from any MBES channel

MBES_1 – generate analysis derived from only MBES channel 1



online_ami – waveform display

MBES - generate analysis derived from any MBES channel

MBES_1 – generate analysis derived from only MBES channel 1

The screenshot shows the MBES online_ami waveform display interface. The main window is titled "MBES (on daq-xpp-ana01)". It features a "Control" panel on the left with buttons for "Run", "Single", and "Processed", along with a "Rate(Hz)" input set to 2.5. Below this is a "Channels" section with checkboxes and colored buttons for ChA (blue), ChB (red), ChC (green), and ChD (magenta). ChA, ChB, and ChC are checked. At the bottom of the control panel are "Edges" and "Cursors" buttons. The main plot area shows a waveform with a vertical scale from -0.03 to 0.02. A smaller window titled "MBES_ChB (on daq-xpp-ana01)" is overlaid on the plot, showing settings for "Chan 1". This window includes a "Plot" section with options for "Normalize to" (a text box), "None", "Single", "Averaged" (selected), "Math" (with an "Expr" text box), and "Reference". The "Events" input is set to 10. There are "Enter", "Load", "Filter", "Y Transform", "OK", and "Close" buttons. Annotations with arrows point to various elements: "Setup initial analysis of MBES waveform" points to the "Channels" section; "Choose which Acqiris channel to analyze" points to the "Chan 1" dropdown; "Normalize to a scalar" points to the "Normalize to" text box; "Single event waveforms" points to the "Single" radio button; "Averaged over N events" points to the "Averaged" radio button and the "Events" input; "Math with constants, scalars, & waveforms" points to the "Math" radio button and the "Expr" text box; "Load a reference waveform" points to the "Load" button; "Define an event filter" points to the "Filter" button; and "Transform y coordinate for this waveform" points to the "Y Transform" button.

Control

Run

Single

Rate(Hz) 2.5

Processed

Channels

✓ ChA

✓ ChB

✓ ChC

ChD

Edges

Cursors

Plot

File Annotate

0.02

0.01

0

-0.01

-0.02

-0.03

0

MBES_ChB (on daq-xpp-ana01)

Chan 1

Plot

Normalize to

Enter

None

Single

Averaged

Events 10

Math Expr

Enter

Reference

Load

Filter

Y Transform

OK

Close

Choose which Acqiris channel to analyze

Normalize to a scalar

Single event waveforms

Averaged over N events

Math with constants, scalars, & waveforms

Load a reference waveform

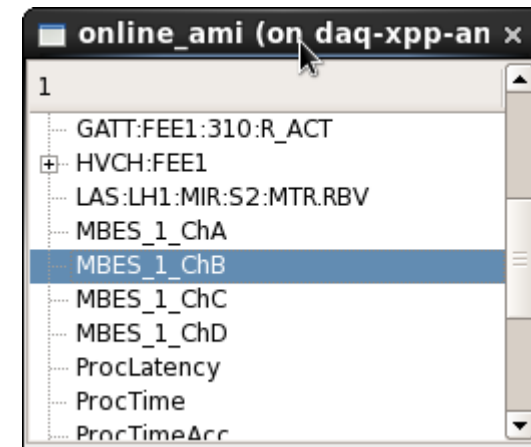
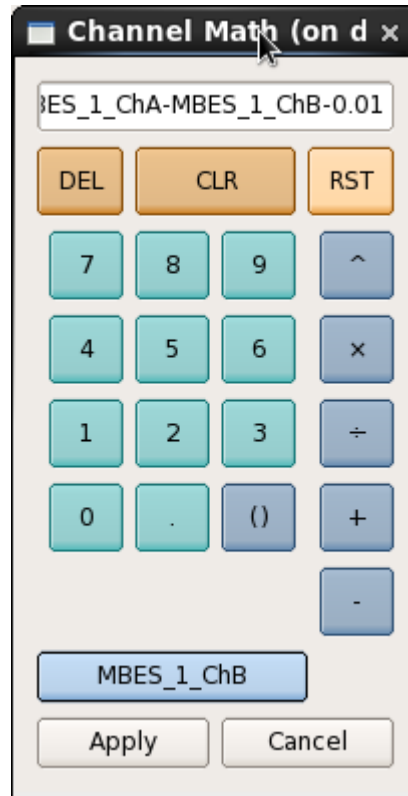
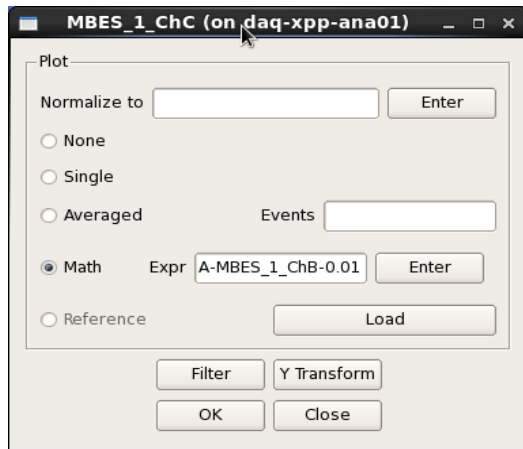
Define an event filter

Transform y coordinate for this waveform

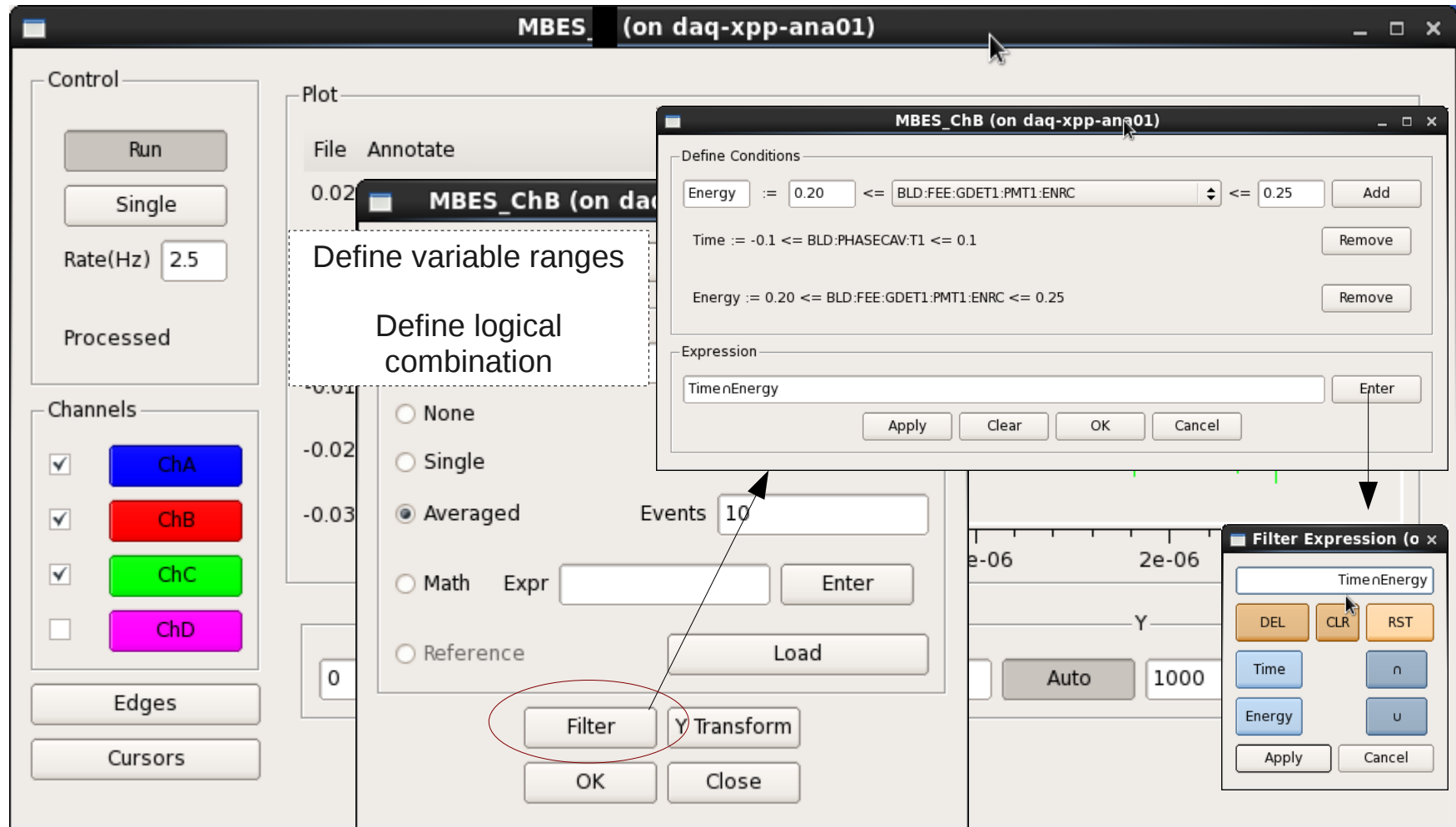
Setup initial analysis of MBES waveform

online_ami – waveform display

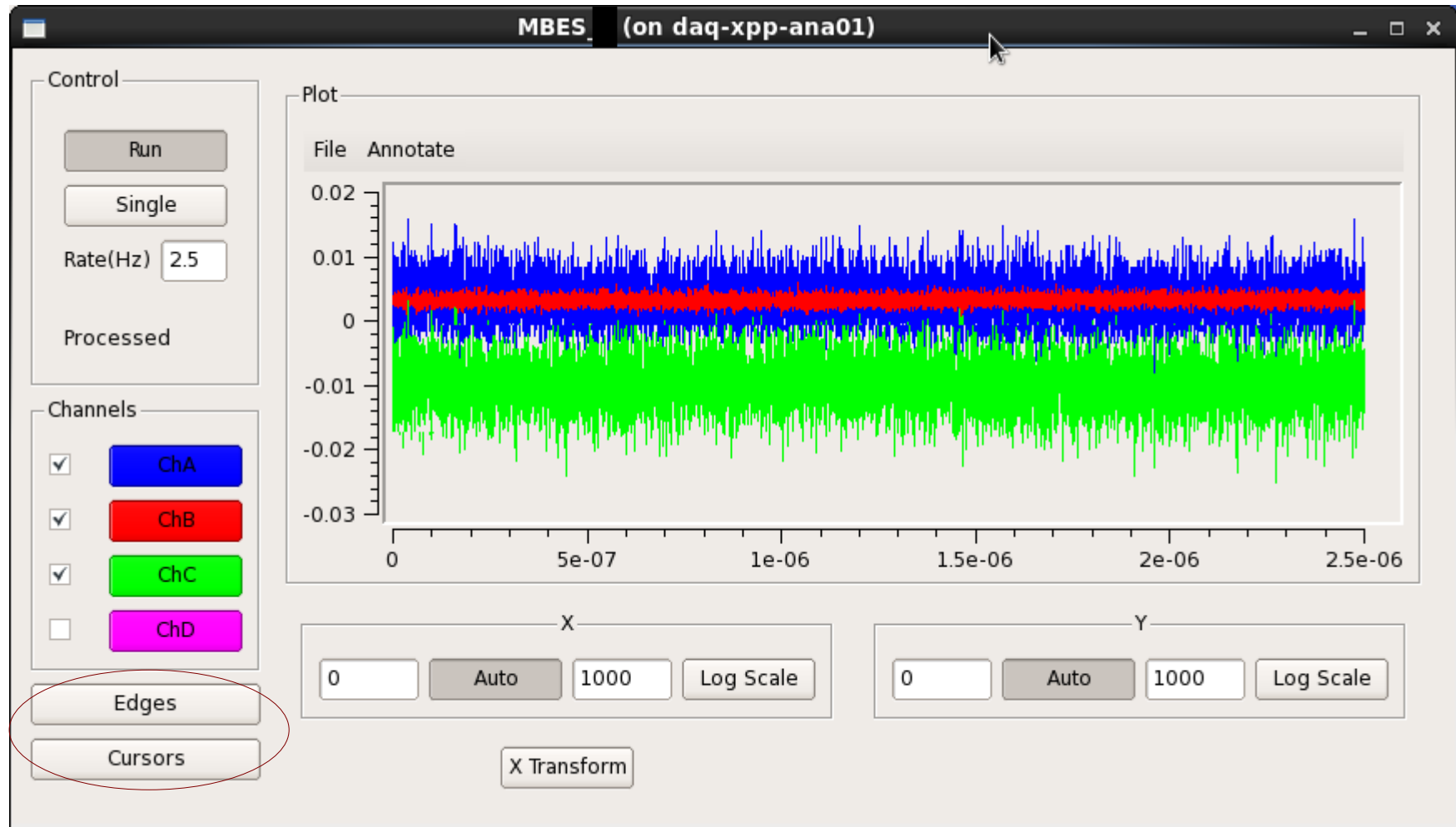
Channel math example:
 $\text{ChA} - \text{ChB} - 0.01(\text{V})$



online_ami – waveform display



online_ami – waveform display



online_ami – waveform edge finder

EdgeFinder Plot (on daq-xpp-ana01)

Locates leading edge of each pulse above 'threshold' at a construct fraction of peak height from 'baseline'.

Source Channel

Channel

Define EdgeFinder

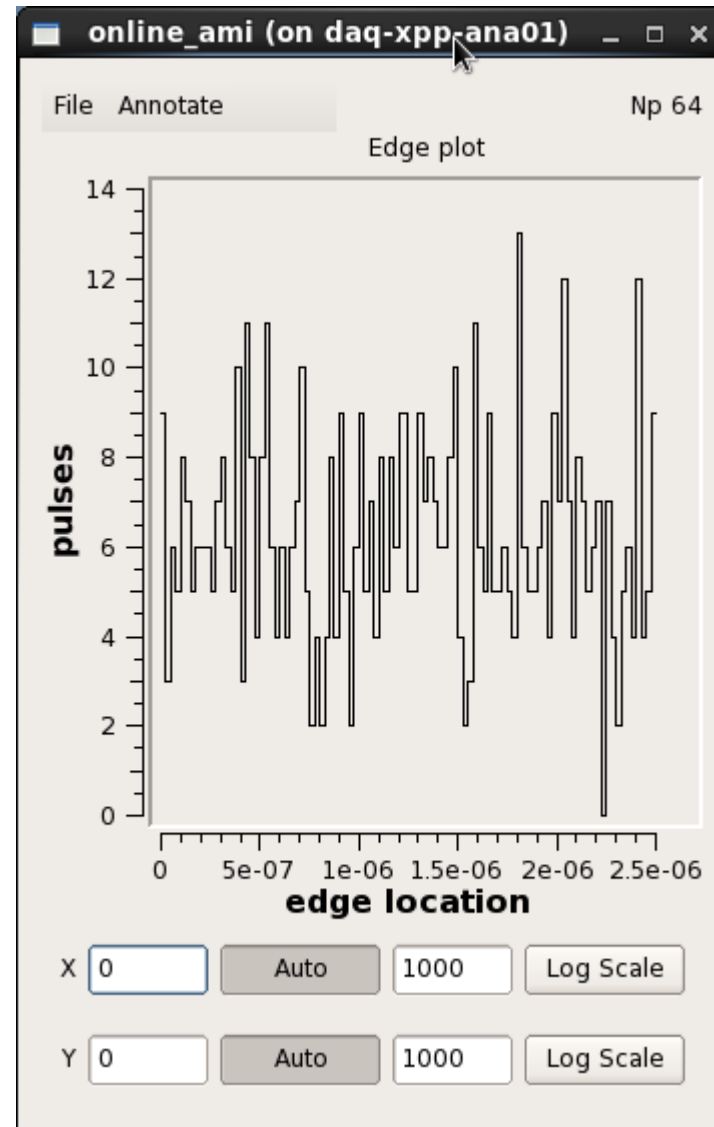
baseline

threshold

Plot

Title

☒ Sum (1dH) bins lo hi



online_ami – waveform cursors

Analyze waveform contents

Define locations on waveform

Enter calculations with those locations

CursorsX Plot (on daq-xpp-ana01)

Source Channel
Channel: MBES_1_ChA

Define CursorsX
Location: 8.27616e-07 [Grab]
a @ 4.6741e-07 [Show] [Delete]
b @ 8.27616e-07 [Show] [Delete]

Expression:
Expr: a/b [Enter]

Plot Type
Title: title [Post]
☐ Sum (1dH) bins: 100 lo: 0 hi: 0
☒ Mean v Time points: 100 prescale: 1
☐ Mean v Var [X Var] bins: 100 lo: 0 hi: 1
☐ Mean v Scan [pts: 200]

Normalization
Normalize ☐ X ☐ Y variable to: []

Weighted Average
☐ Weight by: []

[Plot] [Close]

Cursor Math (on daq-xpp-an x

Expr: a/b

DEL CLR RST

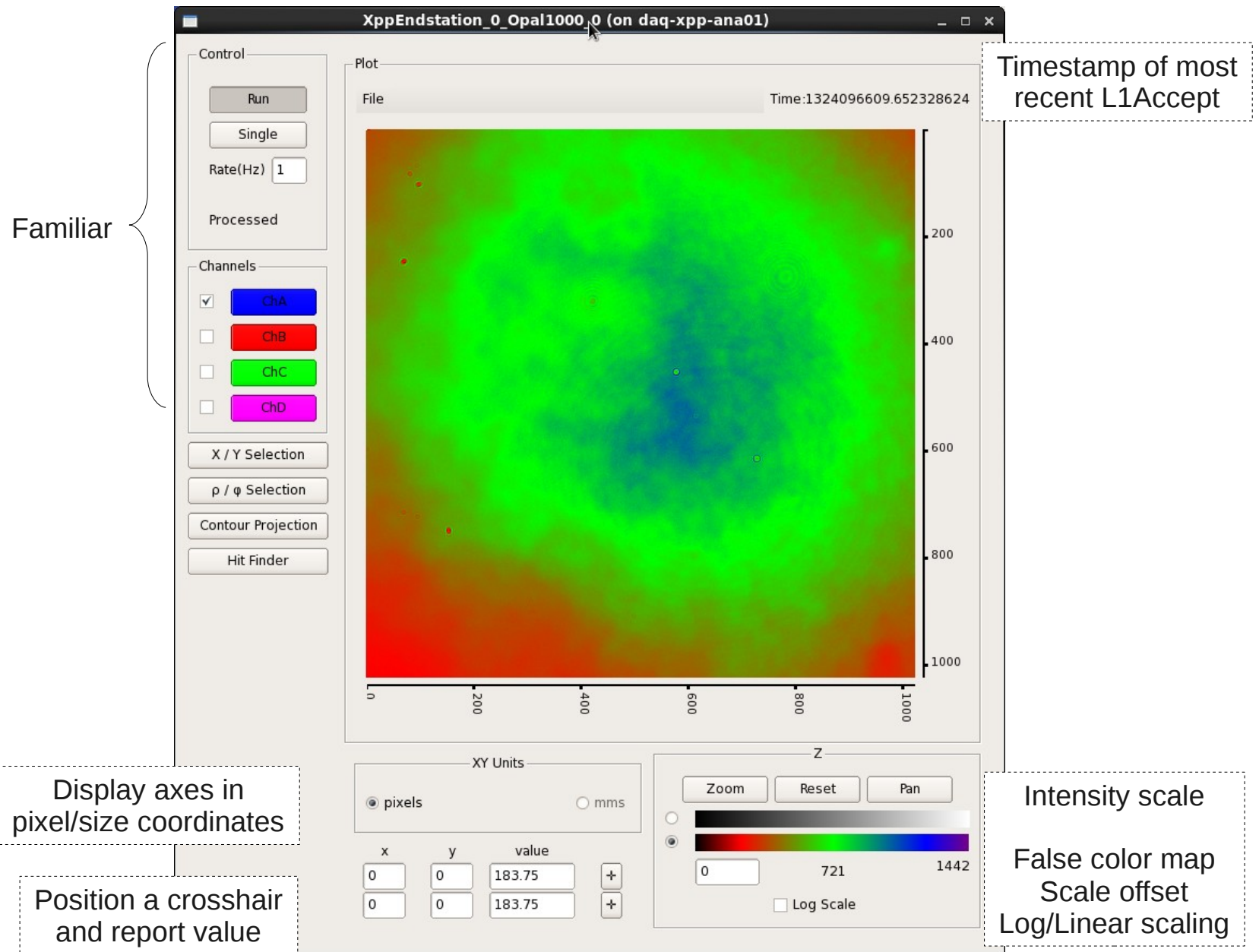
a 7 8 9 ∫ ^
b 4 5 6 ↔ ×
1 2 3 ÷
0 . () +
-

[Apply] [Cancel]

Integrate btw cursors

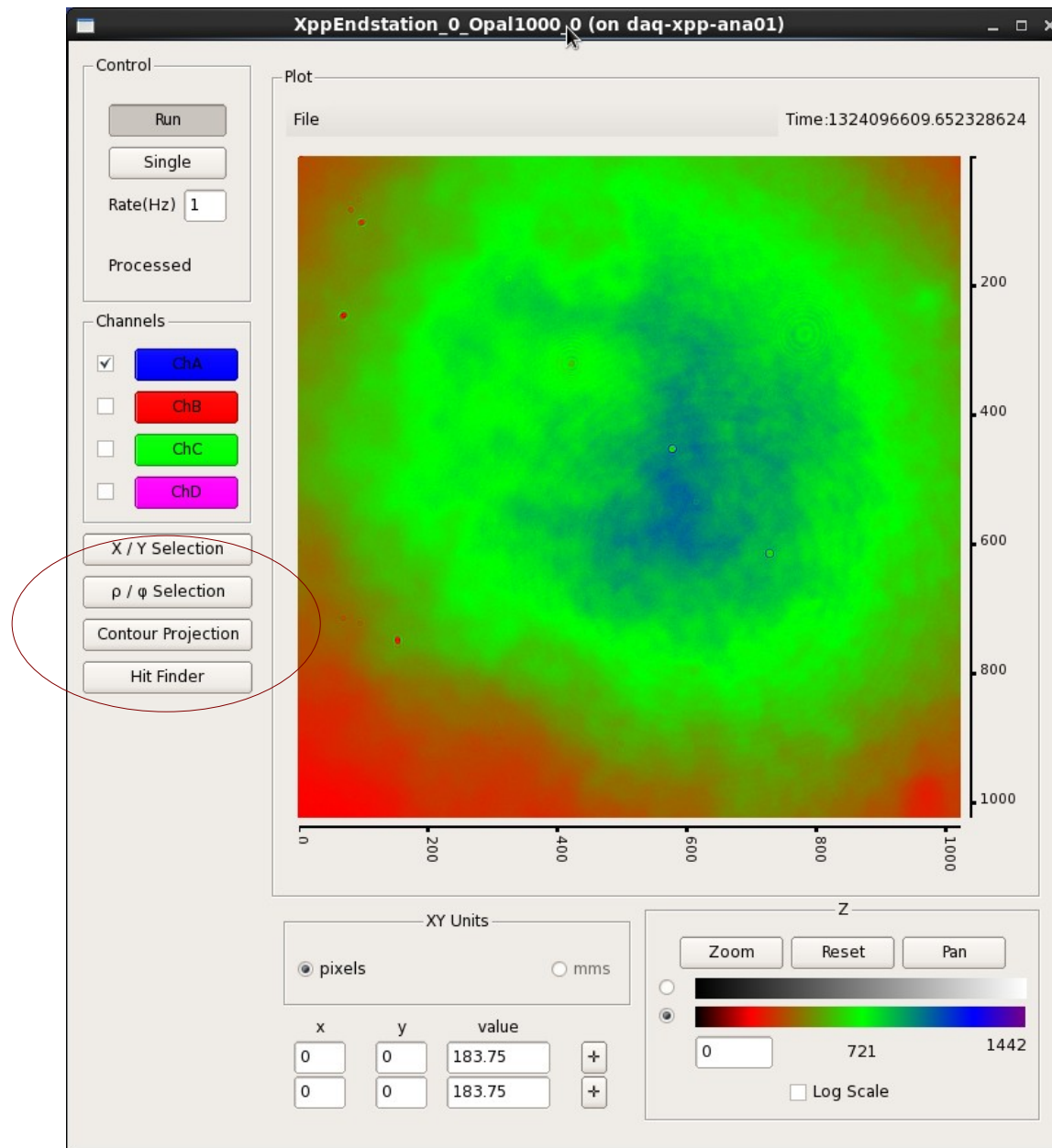
samples btw cursors

online_ami – image display

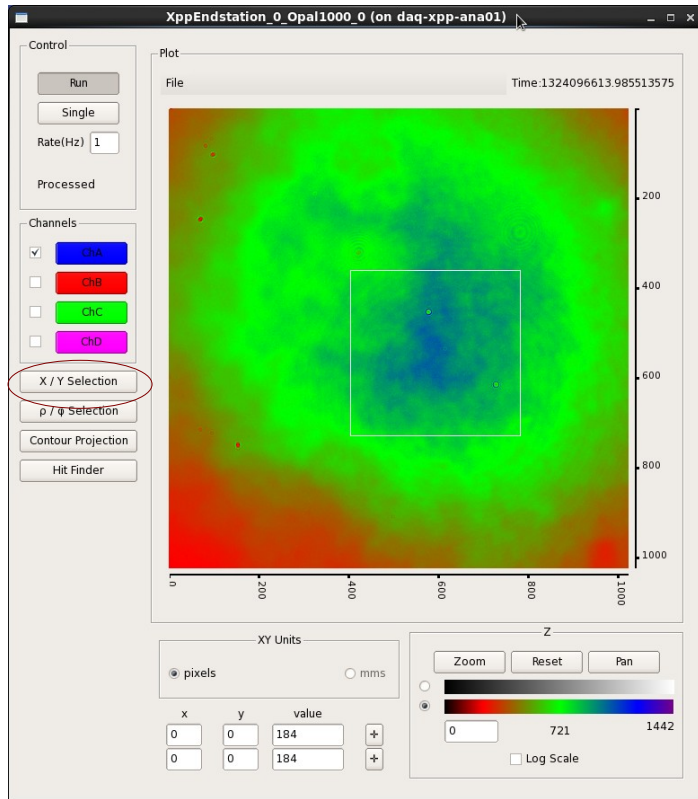


online_ami – image display

Additional
processing

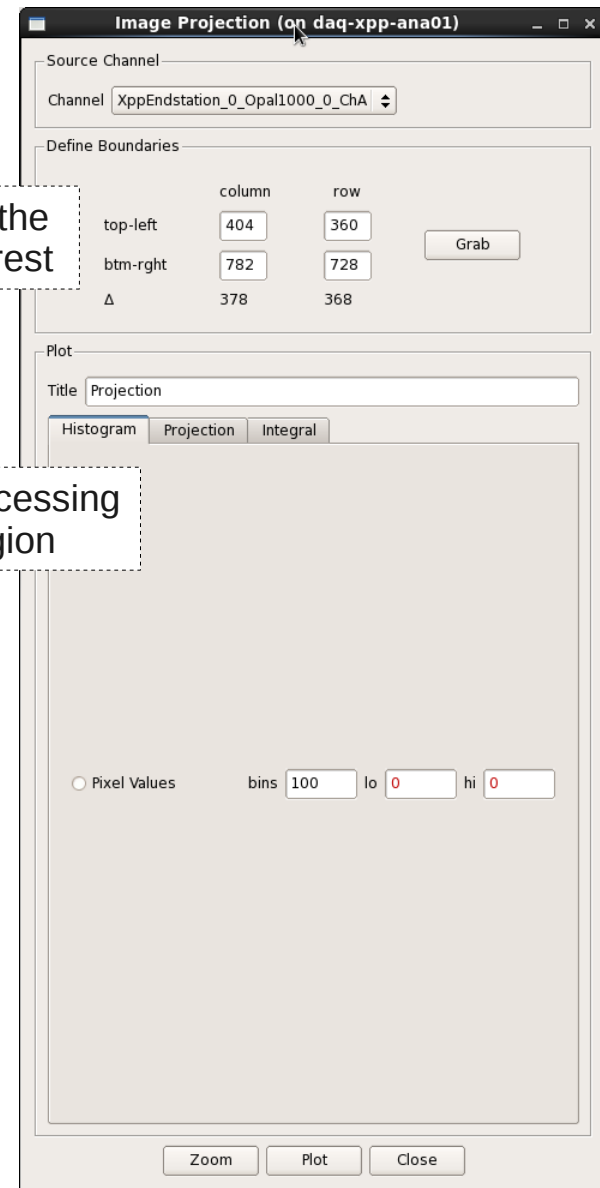


online_ami – image xy projection



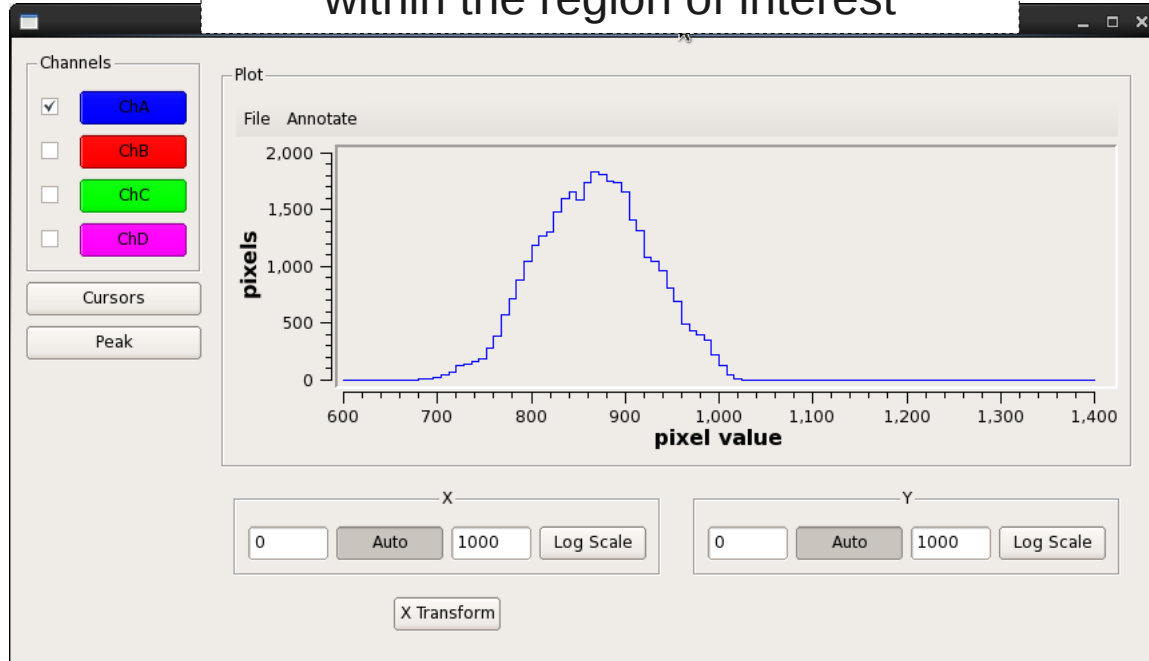
Enter or grab the
region of interest

Select the processing
on that region

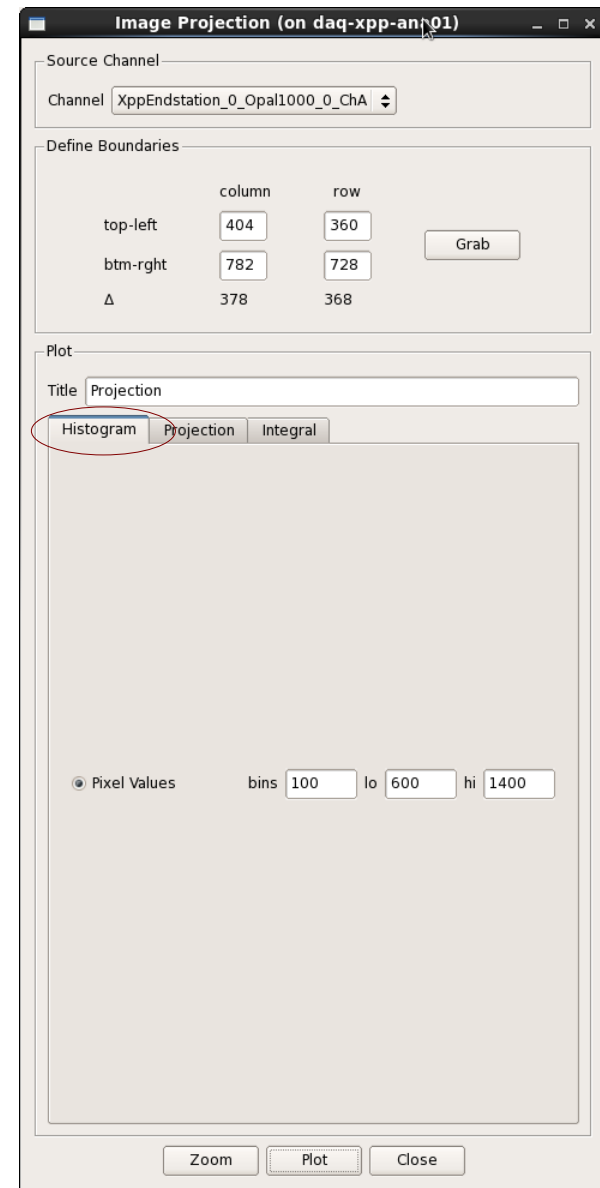


online_ami – image xy projection

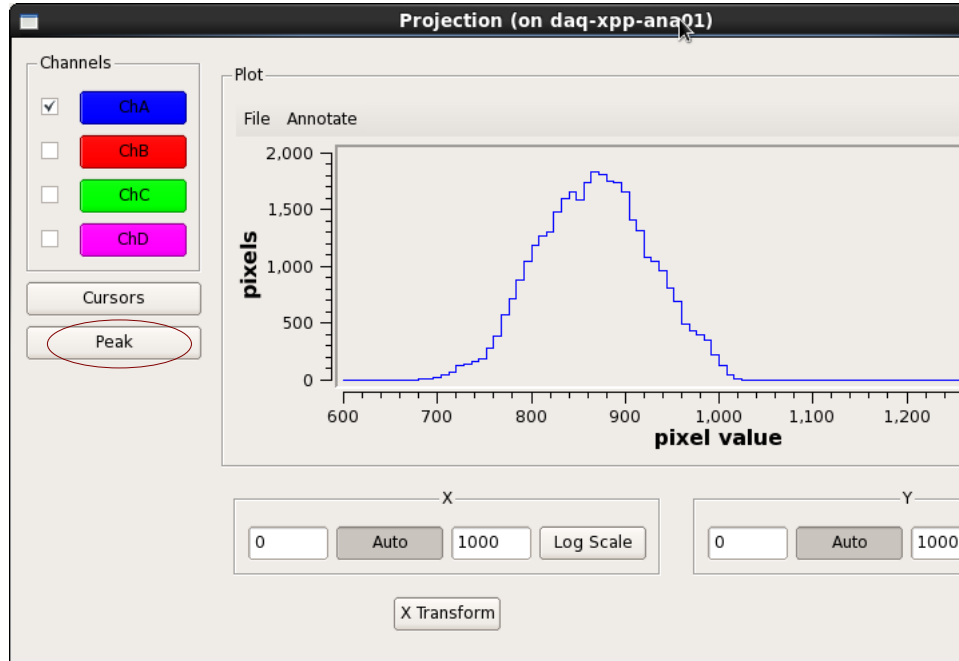
Histogram distribution of pixel values within the region of interest



Generates a 1d projection from which further processing can be done



online_ami – image xy projection



PeakFit Plot (on daq-xpp-ana01)

Finds position, height, and width of largest peak from 'baseline'.

Source Channel

Channel ChA

Define PeakFit

baseline

Grab

Show

Quantity

Position

Plot

Title Peak plot

Post

☒ Sum (1dH) bins 100 lo 0 hi 0

☐ Mean Mean v Time points 100 prescale 1

☐ Mean v Var X Var bins 100 lo 0 hi 1

☐ Mean v Scan pts 200

Plot

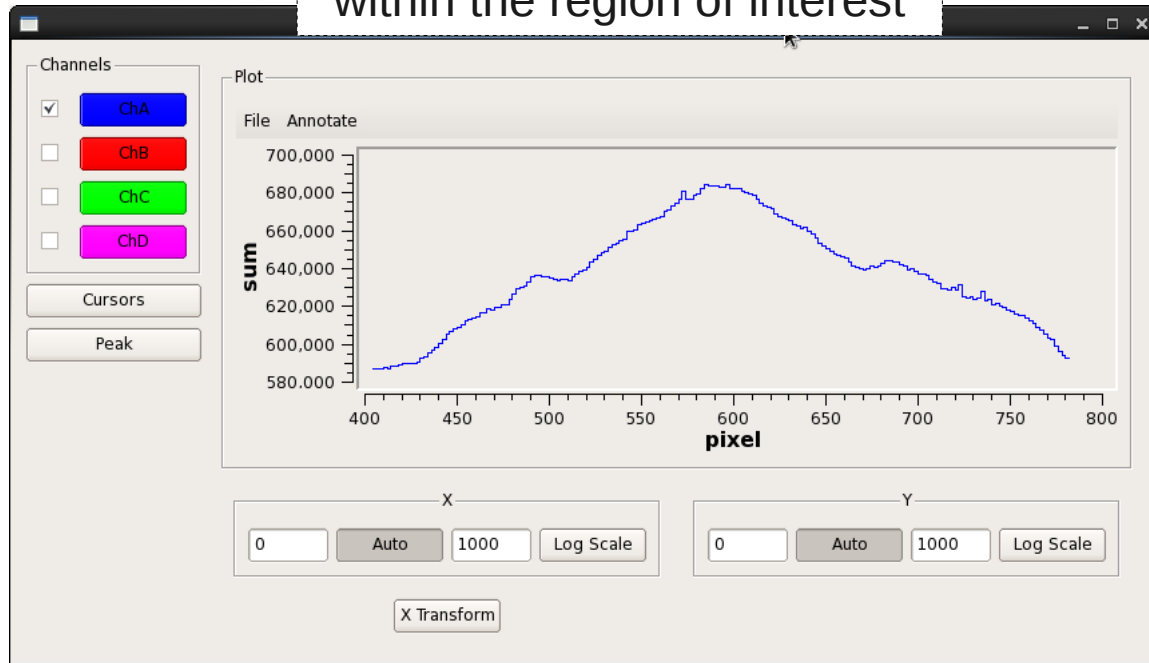
Close

Find peak position,
height, or width

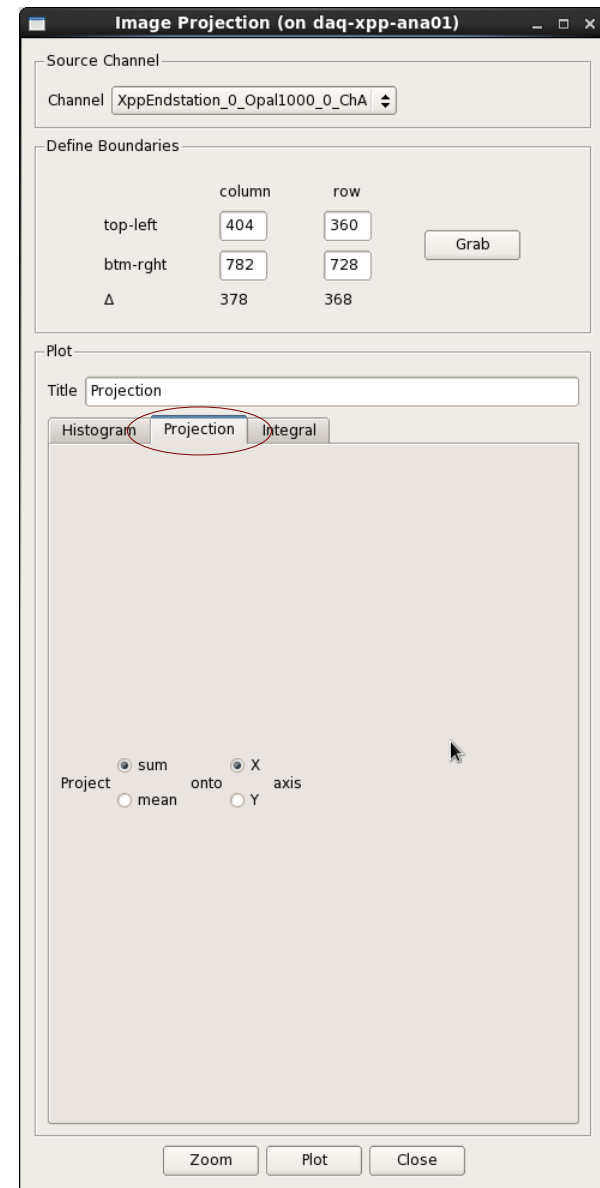
Generate a plot
from that result

online_ami – xy image projection

Integrate pixel rows/columns within the region of interest



Generates a 1d projection from which further processing can be done



online_ami – xy image projection

Integrate all pixel values
within the region of interest

Generates a scalar input
to familiar set of plots

Image Projection (on daq-xpp-ana01)

Source Channel
Channel: XppEndstation_0_Opal1000_0_ChA

Define Boundaries

	column	row
top-left	404	360
btm-right	782	728
Δ	378	368

Grab

Plot
Title: Projection

Histogram Projection **Integral**

Plot Type

Title: title Post

☒ Sum (1dH) bins: 100 lo: 0 hi: 0

☐ Mean v Time points: 100 prescale: 1

☐ Mean v Var X Var bins: 100 lo: 0 hi: 1

☐ Mean v Scan pts: 200

Normalization

Normalize ☐ X ☐ Y variable to:

Weighted Average

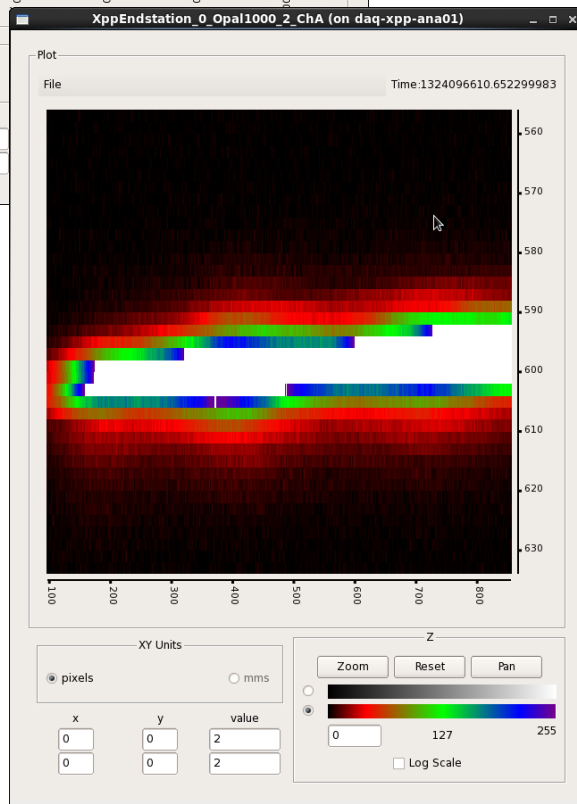
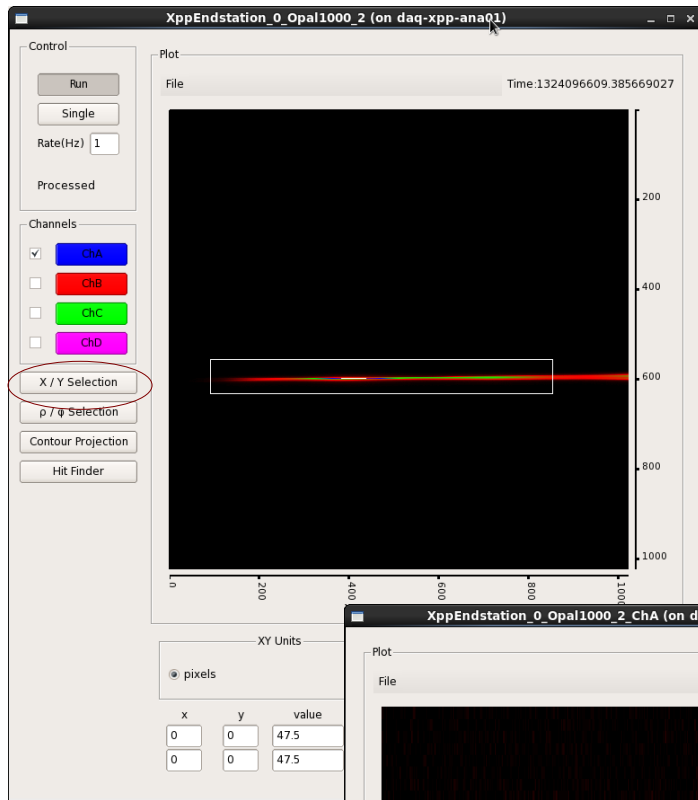
☐ Weight by:

Range Expression

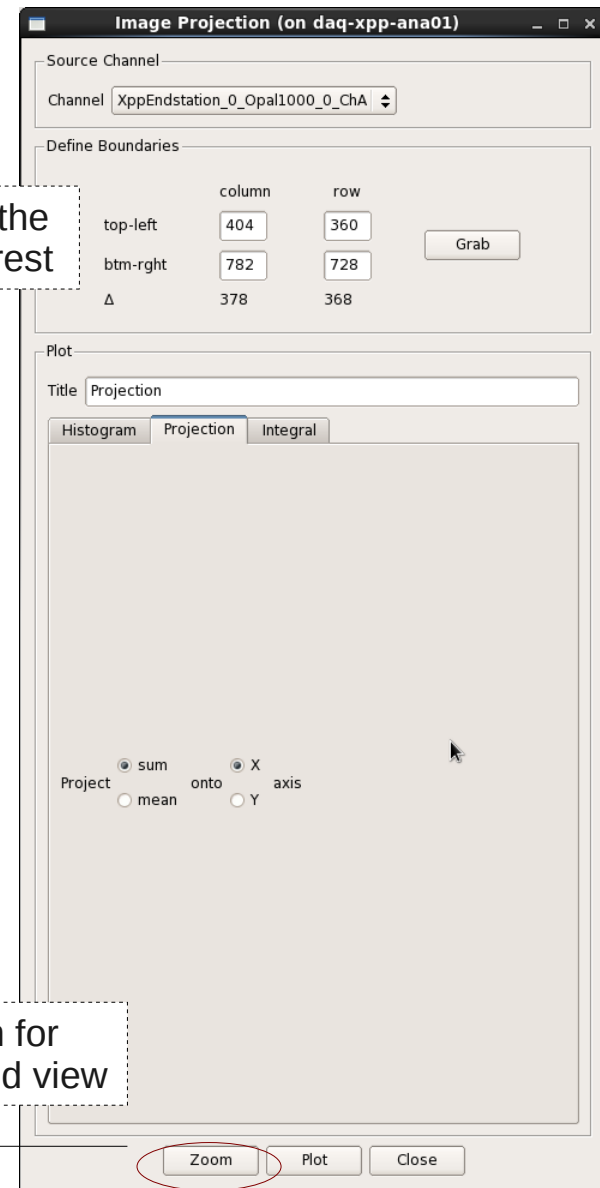
[202],[391][180],[364]

Zoom Plot Close

online_ami – image xy zoom

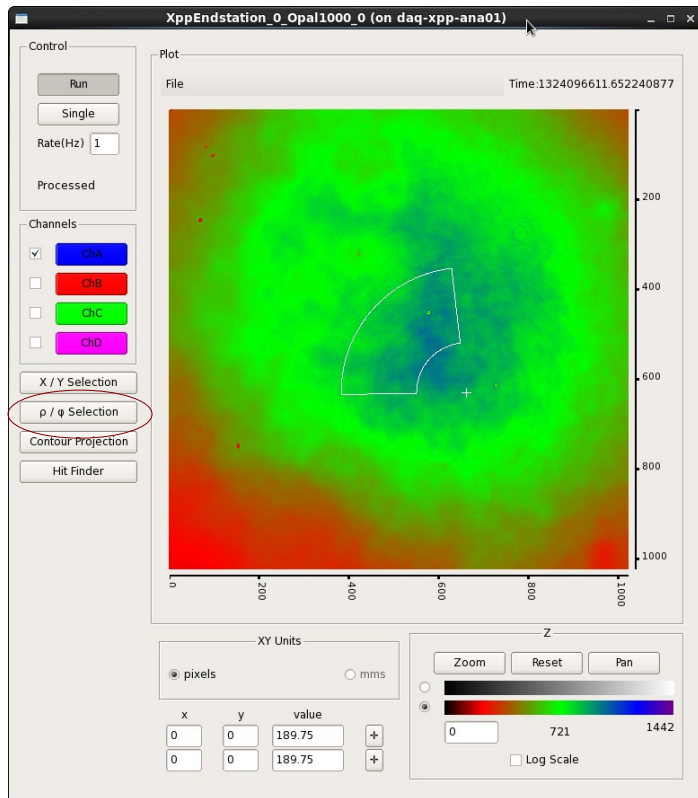


Enter or grab the
region of interest



Zoom for
magnified view

online_ami – image $r\phi$ projection

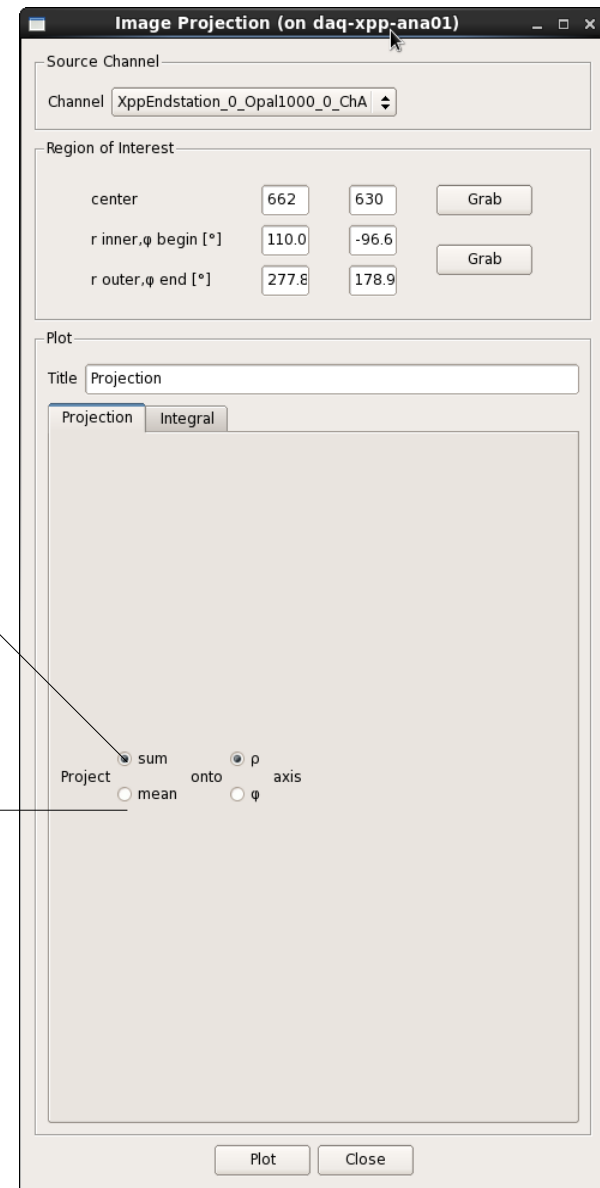


Enter or grab the
region of interest

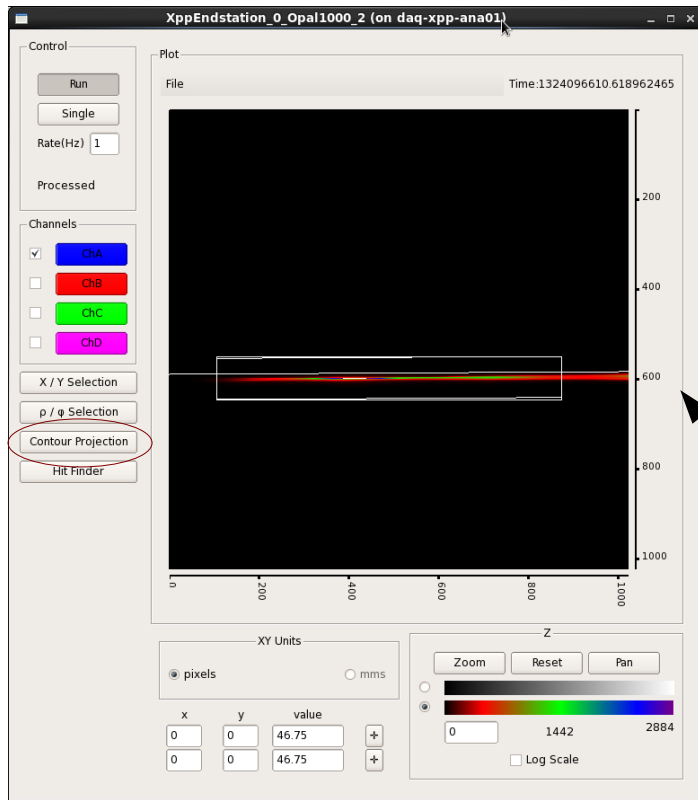
Select the processing
on that region



online_ami – image $r\phi$ projection

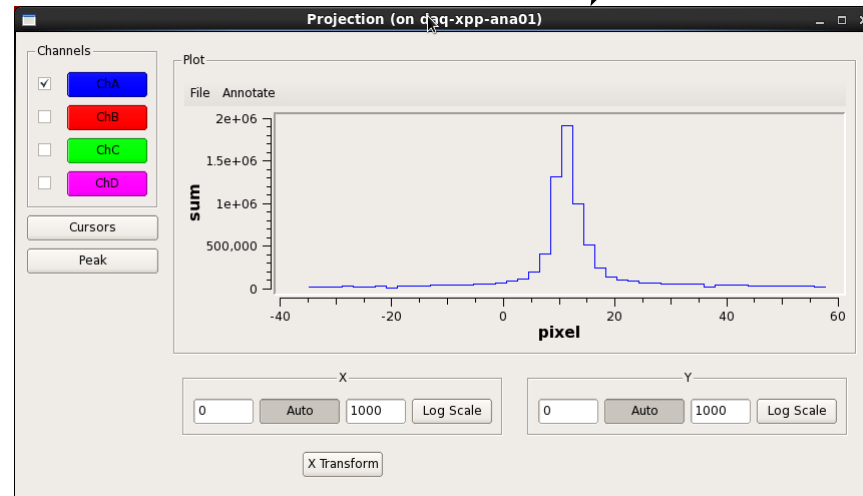
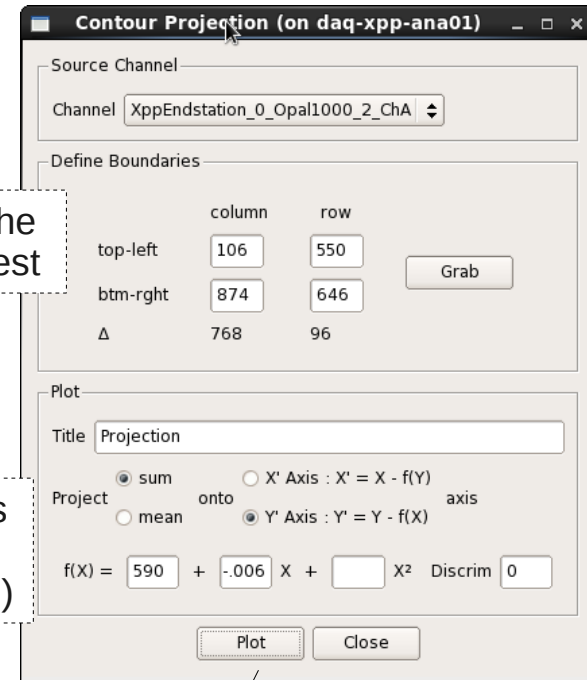


online_ami – image contour projection

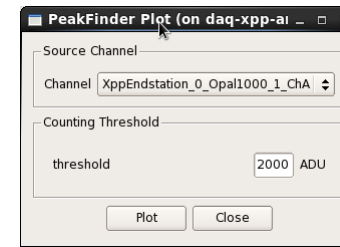
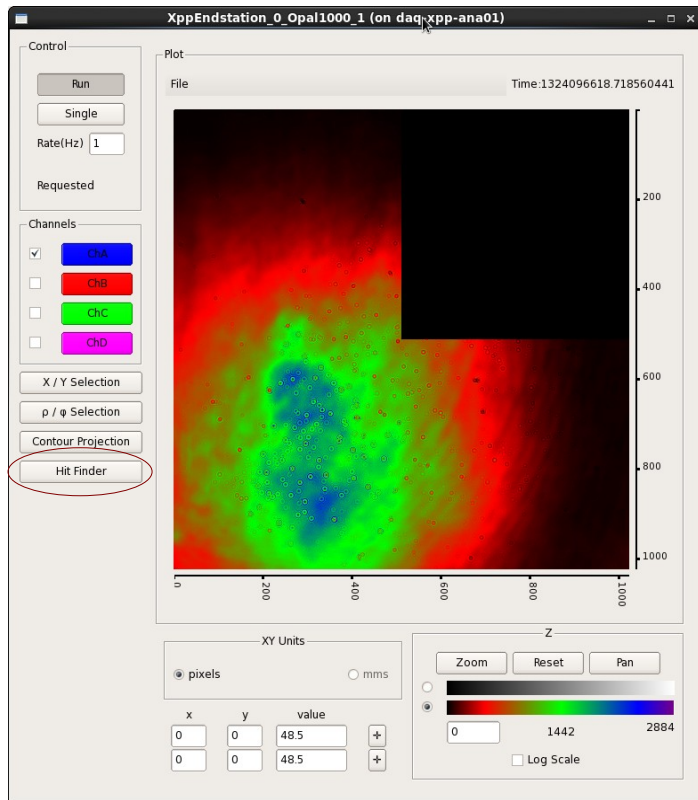


Enter or grab the region of interest

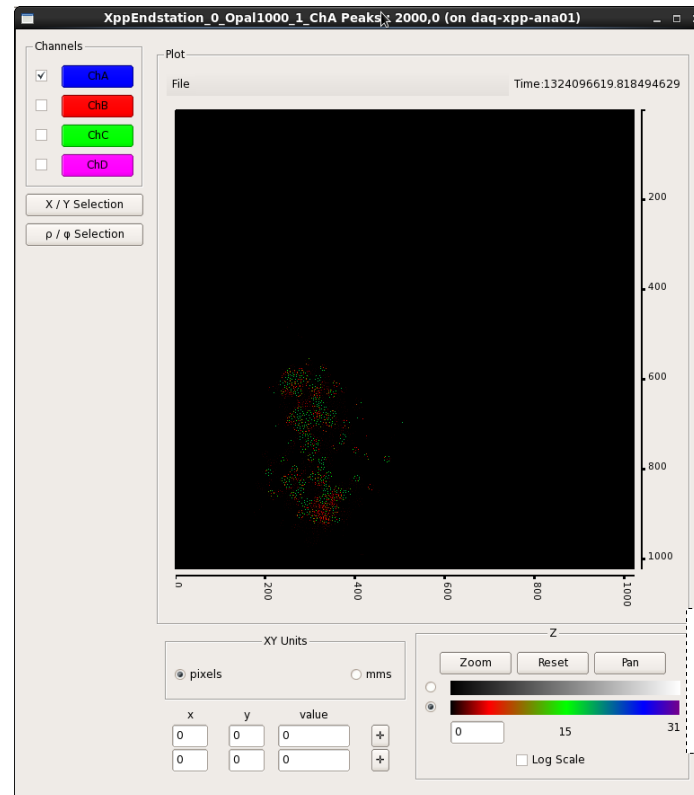
Enter zero-axis function (note white line)



online_ami – image hit finder



Map locations of
local maxima
above *threshold*



Intensity scale
is hits found
(cumulative)

ami development

Adding new detectors

Handlers

class event/*DetectorHandler*

Instantiated in XtcClient
during *configure* transition

On L1Accept
generates prototype *plot* from
xtc data

Adding new plot types

Plot Types

enum data/DescEntry::Type
{Scalar, TH1F, TH2F, Prof,
Image, Waveform, Scan, Ref}

class data/*EntryType*

Deserialized by data/EntryFactory

Operations

enum data/AbsOperator::Type
{Single, Average, XYProjection,
EdgeFinder, EntryMath, BinMath, ...}

class data/*Operation*

Deserialized by data/OperatorFactory

Adding new processing

Analysis Inputs

class qt/*Operation*

Online Analysis Design – *ami* plugins

Use dynamically linked library to extend server functions at runtime

- hutch specific
- reproducible layout
- user controlled code

Plug-in interface includes

- direct access to XTC contents (callbacks for each container)
- global event filter decision
- add scalar variables to global list (“Env”)
- generate plots with a described layout

Difficulties

- too much investment for visitors
- base code evolution

Online Analysis Design – *pyami*

Python library for configuring *ami* server

User scripts replace GUI (manual inputs)

User scripts responsible for plot display

Complementary to *pydaq* – python library for DAQ control

fin