



areaDetector: A module for EPICS area detector support

Mark Rivers

GeoSoilEnviroCARS, Advanced Photon Source

University of Chicago

areaDetector Talk Outline

- Motivation & goals for areaDetector module
- Overview of architecture
- Drivers for detectors & cameras
- Plugins for real-time processing
- Viewers and other clients
- Demo with simDetector and Prosilica camera

areaDetector - Motivation

- 2-D detectors are essential components of synchrotron beamlines
 - Sample viewing cameras, x-ray diffraction and scattering detectors, x-ray imaging, optical spectroscopy, etc.
- EPICS is a very commonly used control system on beamlines, (APS, DLS, SLS, SLAC, NSLS-II, Shanghai, etc.)
- Need to control the detectors from EPICS (useful even on non-EPICS beamlines, since other control systems like SPEC etc. can talk to EPICS)
- Previously several packages available, each typically restricted to a small set of detectors (Flea, Pilatus, marCCD, etc.)
- Clear advantages to an architecture that can be used on any detector, re-using many software components
- Providing EPICS control allows any higher-level client to control the detector and access the data (CSS, SPEC, medm, Python scripts, IDL programs, etc)

areaDetector - Goals

- Drivers for many detectors popular at synchrotron beamlines
 - Handle detectors ranging from >500 frames/second to <1 frame/second
- Basic parameters for all detectors
 - E.g. exposure time, start acquisition, etc.
 - Allows generic clients to be used for many applications
- Easy to implement new detector
 - Single device-driver C++ file to write. EPICS independent.
- Easy to implement detector-specific features
 - Driver understands additional parameters beyond those in the basic set
- EPICS-independent at lower layers.
- Middle-level plug-ins to add capability like regions-of-interest calculation, file saving, etc.
 - Device independent, work with all drivers
 - Below the EPICS layer for highest performance

areaDetector – Data structures

- **NDArray**
 - N-Dimensional array.
 - Everything is done in N-dimensions (up to 10), rather than 2. This is needed even for 2-D detectors to support color.
 - This is what plug-ins callbacks receive from device drivers.
- **NDArrtribute**
 - Each NDArray has a list of associated attributes (metadata) that travel with the array through the processing pipeline. Attributes can come from driver parameters, any EPICS PV, or any user-written function.
 - e.g. can store motor positions, temperature, ring current, etc. with each frame.
- **NDArrayPool**
 - Allocates NDArray objects from a freelist
 - Plugins access in readonly mode, increment reference count
 - Eliminates need to copy data when sending it to callbacks.

EPICS areaDetector Architecture

Layer 6
EPICS CA clients

Channel Access Clients (medm, IDL, ImageJ, SPEC, etc.)

Layer 5
Standard
EPICS records

ADBase .template xxxDriver .template NDPluginBase .template NDPluginXXX .template

Layer 4
EPICS device
support

Standard asyn device support
(device-independent)

C++ Base classes
(NDArray, asynPortDriver,
asynNDArrayDriver,
ADDriver, NDPluginDriver)

Layer 3
Plug-ins

StdArrays Process ROI File
(netCDF, TIFF, JPEG,
HDF5)

Layer 2
Device drivers

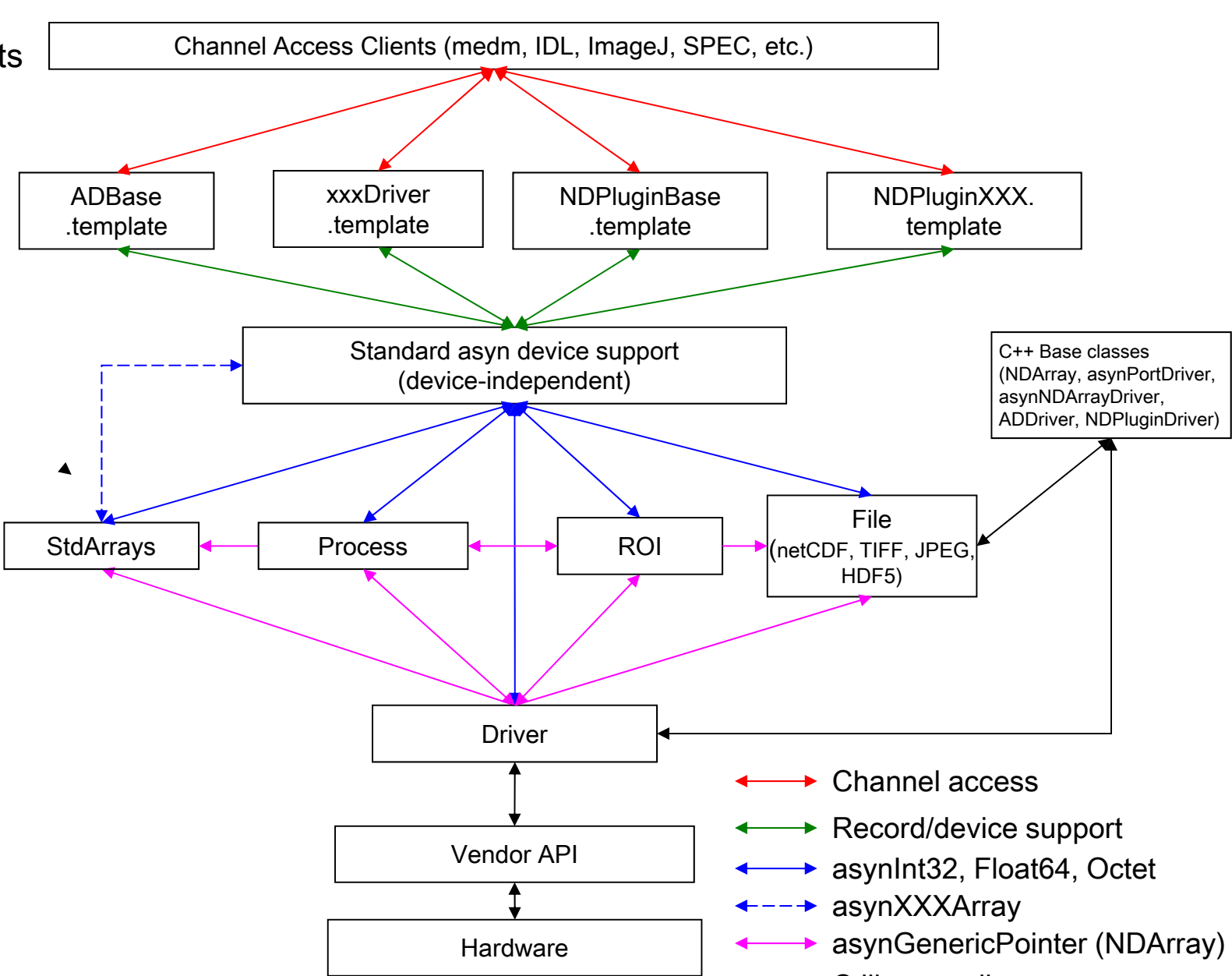
Driver

Layer 1
Hardware API

Vendor API

Hardware

- ↔ Channel access
- ↔ Record/device support
- ↔ asynInt32, Float64, Octet
- - - asynXXXArray
- ↔ asynGenericPointer (NDArray)
- ↔ C library calls



areaDetector – Data structures

Look at NDAArray.h

Look at NDAttribute.h

Look at an XML attribute file

areaDetector Organization

(R2-x and later)

areaDetector

Top-level module
RELEASE files, documentation, Makefile

ADCore

Core module
Base classes, plugins,
simDetector, documentation

ADBinaries

Binary libraries for
Windows (HDF5,
GraphicsMagick)

ADProsilica

Prosilica driver

ADPilatus

Pilatus driver

...

- Each box above is a separate git repository
- Can be released independently
- Hosted at <http://github.com/areaDetector> project
- Each repository is a submodule under areaDetector/areaDetector

Source Code Organization on github

- <https://github.com/areaDetector> is top-level project
- Contains configure/ directory where paths and versions of supporting software are defined
- Contain .gitmodules to define submodules that will be cloned with `git clone --recursive`
- Contains documentation directory that builds and installs documentation
- Contains a top-level Makefile to build all or selected submodules

Detector drivers

- ADDriver (in ADCore)
 - Base C++ class from which detector drivers derive. Handles details of EPICS interfaces, and other common functions.
- Simulation driver (in ADCore)
 - Produces calculated images up to very high rates. Implements nearly all basic parameters, including color. Useful as a model for real detector drivers, and to test plugins and clients.
- Prosilica driver (ADProsilica)
 - Gigabit Ethernet cameras, mono and color
 - High resolution, high speed, e.g. 1360x1024 at 30 frames/second = 40MB/second.
- Firewire (IEEE-1396 DCAM) (ADFireWireWin, firewireDCAM)
 - Vendor-independent Firewire camera drivers for Linux and Windows
- Roper driver (ADRoper)
 - Princeton Instruments and Photometrics cameras controlled via WinView

Detector drivers (continued)

- PVCAM driver (ADPvCam)
 - Princeton Instruments and Photometrics cameras controlled via PVCAM library
- Pilatus driver (ADPilatus)
 - Pilatus pixel-array detectors.
- marCCD driver (ADmarCCD)
 - Rayonix (MAR-USA) CCD x-ray detectors
- ADSC driver (ADADSC)
 - ADSC CCD detectors
- mar345 driver (ADmar345)
 - marResearch mar345 online image plate
- Perkin-Elmer driver (ADPerkinElemer)
 - Perkin-Elmer amorphous silicon flat-panel detectors

Detector drivers (continued)

- Bruker driver (ADBruker)
 - Bruker detectors controlled via their Bruker Instrument Server (BIS)
- LightField driver (ADLightField)
 - Princeton Instruments detectors controlled via their LightField application using the Microsoft Common Language Runtime to automate it
- PSL driver (ADPSL)
 - Photonic Sciences Limited detectors
- URL driver (ADURL)
 - Driver to display images from any URL. Works with Web cameras, Axis video servers, static images, etc.
- Andor driver (ADAndor)
 - Driver for Andor CCD cameras
- Andor3 driver (ADAndor3)
 - Driver for Andor sCMOS cameras with V3 of their SDK

Detector drivers (continued)

- Point Grey driver (ADPointGrey)
 - Driver for GigE, USB-3.0, USB-2.0, and Firewire cameras from Point Grey Research
- Pixirad driver (ADPixirad)
 - Driver for CdTe pixel-array detectors from Pixirad
- Generic GigE driver (aravisGigE)
 - Should work with any GigEVision compliant camera. From Tom Cobb at Diamond. Uses Aravis reverse-engineered GigEVision library
- PVAccess (EPICS V4) driver
 - Receives NTNDArrays over PVAccess
 - Allows plugins to run on other processes or machines from the areaDetector driver

ADBase.adl – Generic control screen

- Works with any detector
- Normally write custom control for each detector type to hide unimplemented features and expose driver-specific features

ADBase.adl

Area Detector Control - 13SIM1:cam1:

Setup

asyn port SIM1
EPICS name 13SIM1:cam1:
Manufacturer Simulated detector
Model Basic simulator
Connected
Connection
More

Shutter

Shutter mode
Status: Det. Closed EPICS Closed
Open/Close
Delay: Open 0.000 Close 0.000
EPICS shutter setup

Collect

Exposure time 0.010 0.010
Acquire period 0.000 0.000
Images 10 10
Images complete 703
Exp./image 1 1
Image mode Continuous
Trigger mode Internal
Acquire
Detector state Collecting
Time remaining 0.000
Image counter 0 703
Image rate 67.0
Array callbacks Enable

Readout

	X	Y
Sensor size	640	480
Binning	<input type="button" value="1"/>	<input type="button" value="1"/>
Region start	<input type="button" value="0"/>	<input type="button" value="0"/>
Region size	<input type="button" value="640"/>	<input type="button" value="480"/>
Reverse	<input type="button" value="No"/>	<input type="button" value="No"/>
Image size	640	480
Image size (bytes)	307200	
Gain	<input type="button" value="1.000"/>	<input type="button" value="1.000"/>
Data type	<input type="button" value="UInt8"/>	<input type="button" value="UInt8"/>
Color mode	<input type="button" value="Mono"/>	<input type="button" value="Mono"/>

File

Driver file I/O

Pilatus specific control screen

The screenshot displays the 'Pilatus Detector Control - 13PIL1:cam1' window, which is organized into several functional panels:

- Setup:** Contains configuration fields for 'asyn port PIL', 'EPICS name 13PIL1:cam1', 'Manufacturer Dectris', and 'Model Pilatus'. It shows a 'Connected' status and includes 'Connect', 'Disconnect', and 'Debugging' buttons.
- Shutter:** Features a 'Shutter mode' dropdown set to 'None', 'Status: Det. Closed' and 'EPICS Closed', 'Open/Close' buttons, and 'Delay: Open 0.000' and 'Close 0.000' fields. An 'EPICS shutter setup' button is also present.
- Collect:** Includes 'Exposure time 1.000', 'Acquire period 0.150', '# Images 1', 'Delay time 0.000000', '# Exp./image 1', and 'Trigger mode Internal'. It has 'Acquire Start/Stop' buttons, 'Armed Unarmed' status, 'Image counter 0', 'Image rate 0.0', and 'Array callbacks Enable' options.
- Status:** Shows 'Status: Waiting for acquire command', 'To camserver: Exposure /corvette/home/epics/temp/pilatus_test_A_081.tif', and 'From camserver: 7 OK /corvette/home/epics/temp/pilatus_test_A_081.tif'.
- Data corrections:** Includes 'Bad pixel file', '# Bad pixels: 0', 'Flat field file', and 'Flat field valid: No' with a 'Min. flat field: 100' field.
- File:** Displays the current file path '/corvette/home/epics/temp/pilatus_test_A', file name 'pilatus_test_A', 'Next file # 82', 'Auto increment Yes', and 'Filename format %s_%3.3d.tif'. It also shows 'File format TIFF' and 'Last filename /corvette/home/epics/temp/pilatus_test_A_081.tif'.
- Attributes:** Shows the file 'pilatusAttributes.xml'.
- Detector:** Lists 'Detector Size 487 195', 'Threshold (keV) 10.000', 'Shaping time/Gain 5-18KeV/Med/MedG', 'Pixel cutoff 1071635', 'Read file timeout 20.000', 'Gap fill N.A.', and environmental data for 'Temperature 0.0 -99.0 0.0' and 'Humidity 0.0 -99.0 0.0'. It also indicates 'TVX version Unknown'.
- Plugins:** Offers 'All', 'File', 'ROI', 'Stats', and 'Other' options.

MAR-345 specific control screen

marCCD.adl

marCCD Detector Control - 14ID_MX340:cam1:

Setup

asyn port **MAR**
EPICS name **14ID_MX340:cam1:**
Manufacturer **MAR**
Model **CCD**
Connected
Connection
Debugging

Shutter

Shutter mode
Status: Det. EPICS
Open/Close
Delay: Open Close
EPICS shutter setup

Collect

Exposure time
Acquire period
images
images counter
Image mode
Frame type
Overlap mode
Trigger mode
Readout mode
Gate mode
Array callbacks
Acquire
Image counter

Attributes

File

Status

Detector state Time remaining
Server state Readout status
Task status Correct status
Acquire status Writing status
Dezinger status Series status
Status poll rate
To marCCD server:
From marCCD server:

File

File path Exists:
File path
File name
Next file #
Auto increment Ancillary information
Filename format Example: %s%s_%3.3d.tif

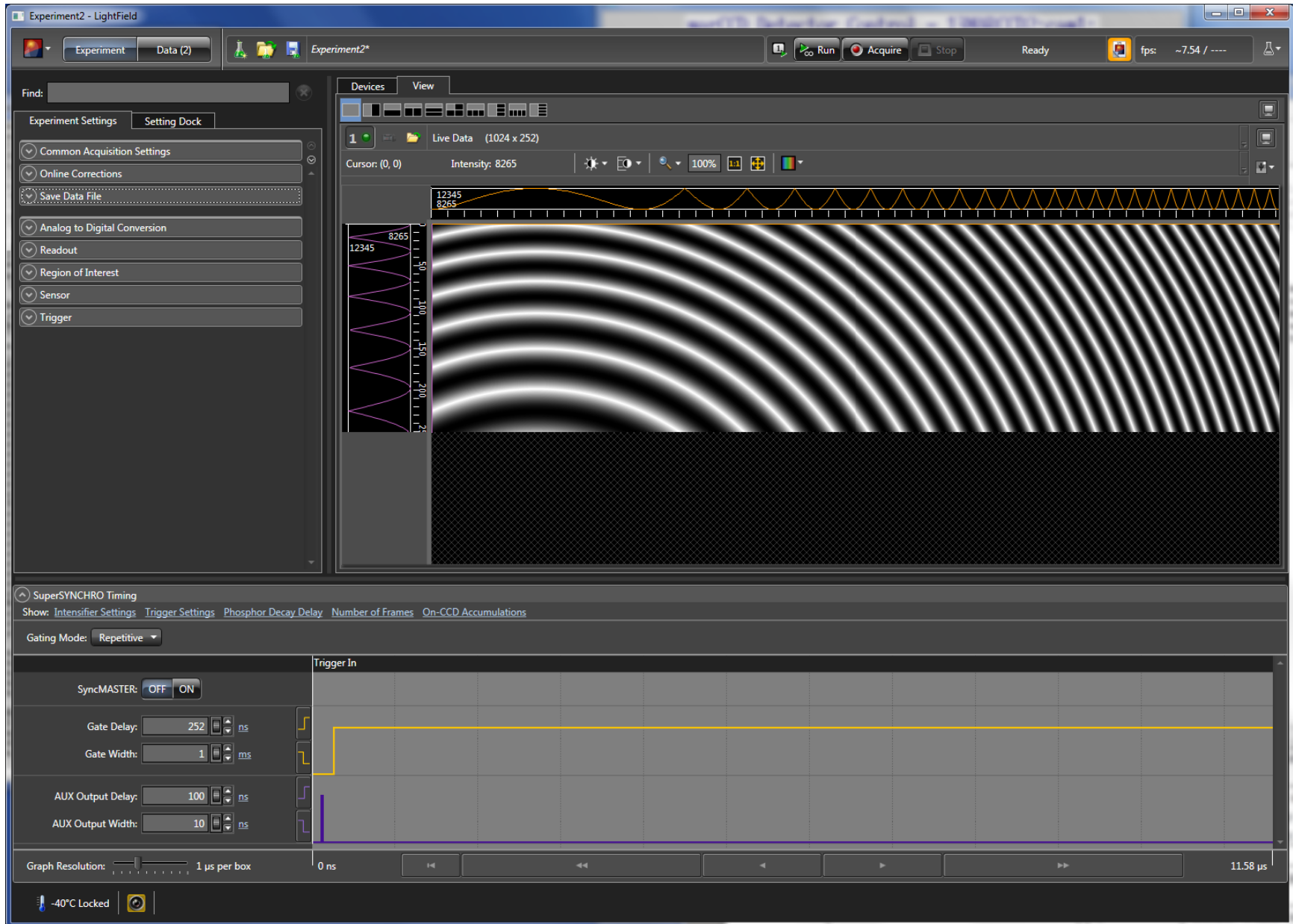
Series format Example: %s%s_%3.3d
First series #
Series digits
Last filename
Save file Auto save

Readout

	X	Y
Detector Size	7680	7680
	2	2
Binning	<input type="text" value="2"/>	<input type="text" value="2"/>
Image Size	3840	3840
Image Size (bytes)	29491200	
Frame shift	<input type="text" value="0"/>	<input type="text" value="0"/>
Stability	<input type="text" value="0.00"/>	<input type="text" value="0.00"/>
Server mode	<input type="text" value="2"/>	

Plugins

LightField driver



LightField driver

LightField.adl

Area Detector Control - 13LF1:cam1:

Setup

asyn port LF1
 EPICS name 13LF1:cam1:
 Manufacturer Princeton Instrument
 Model PIXIS: 100BR
Connected

Connection

Debugging

Shutter

Shutter Type
 LF Shutter Mode
 Status: Det. Closed EPICS Closed
 Open/Close
 Delay: Open 0.000 Close 0.000
 EPICS shutter setup

Spectrometer

[860nm, 300] [1] [0]
 Grating
 Center wavelength 750.000 750.000
 Entrance width 100 100
 Exit port Front

Plugins

Collect

Exposure time 5.000
 Acquire Period 0.000 0.000
 # Accumulations 0 0
 # Exposures 1 1
 # Frames 1 1
 # Exposures Complete 0
 # Frames Complete 1535
 # Acquisitions 0 0
 # Acquisitions Complete 0
 Image Mode Normal
 Trigger Mode
Done
 Acquire
 Detector State Idle
 Ready to Run Ready
 Image counter 1535
 Image Rate 0.0
 Array Callbacks Disable

Intensifier

Int. Enable Disable
 Intensifier Gain 0
 Gating Mode Repetitive
 Trigger Frequency 1e+001 1e+001
 SyncMaster Enable
 SyncMaster2 Delay 1.00e-00 1.00e-004
 Rep. Gate Width 5.00e-00 5.00e-002
 Rep. Gate Delay 0.00e+00 0.00e+00
 Seq. Start Width 0.00e+00 0.00e+00
 Seq. Start Delay 0.00e+00 0.00e+00
 Seq. End Width 0.00e+00 0.00e+00
 Seq. End Delay 0.00e+00 0.00e+00
 Aux I/O Width 2.00e-00 2.00e-006
 Aux I/O Delay 0.00e+00 0.00e+00

Readout

	X	Y
Sensor Size	1340	100
Binning	1	1
Region Start	0	78
Region Size	1340	10
Reverse	No	No
Image Size	1340	10
Image Size (bytes)		26800
Gain	Low	Medium
Data type		UInt16
Temperature	-75.000	-75.000
Actual temperature		-75.000

Experiment

PIXIS 5_29_2013.lfe
 Experiment

Attributes

File

URL Driver

- Driver that can read images from any URL.
- Can be used with Web cameras and Axis video servers.
- Uses GraphicsMagick to read the images, and can thus handle a large number of image formats (JPEG, TIFF, PNG, etc.).

The screenshot shows the URLDriver.adl application window titled "Area Detector Control - 13URL1:cam1:". It features several control panels:

- Setup:** Includes fields for asyn port (URL1), EPICS name (13URL1:cam1), Manufacturer (URL Driver), and Model (GraphicsMagick). It shows a "Connected" status and buttons for "Connect" and "Disconnect".
- Shutter:** Controls shutter mode (set to "None"), status (Det. Closed), and EPICS status (Closed). It has "Open" and "Close" buttons, delay fields (0.000), and an "EPICS shutter setup" button.
- Collect:** Manages acquisition with fields for Acquire period (0.100), # Images (1), and # Images complete (1096). It includes an "Image mode" dropdown (set to "Continuous") and "Start" and "Stop" buttons.
- Readout:** Displays image parameters: X (704), Y (480), Image size (bytes) (1013760), Data type (UInt8), and Color mode (RGB1).
- Attributes:** A "File" field at the bottom.

A status bar at the bottom shows the current URL: "BMC Hutch (Axis) Setup http://164.54.160.141/jpg/1/hugesize.jpg".

The screenshot shows the URLDriverSetup.adl application window titled "URL Setup - 13URL1:cam1:". It contains a table with the following data:

Description	URL
1 BMC Hutch (Axis)	http://164.54.160.141/jpg/1/hugesize.jpg
2 BMC Sample (Axis)	http://164.54.160.141/jpg/2/hugesize.jpg
3 The Sun!	images/sun.jpg
4 marCCD	images/marCCD.tif
5 MultiTIFF	images/MultiTIFF.tiff
6 URL6	
7 URL7	
8 URL8	
9 URL9	
10 URL10	

Andor Driver

- Supports USB and PCI CCD cameras from Andor.
- Runs on 32-bit and 64-bit Linux and 32-bit and 64-bit Windows.
- Original version by Matt Pearson from Diamond Light Source.

The screenshot shows the 'Andor Detector Control - 13ANDOR1:cam1' window. It is divided into several functional panels:

- Setup:** Contains fields for 'asyn port' (ANDOR), 'EPICS name' (13ANDOR1:cam1), 'Manufacturer' (Andor), and 'Model' (DY934_BR, DD). It also shows a 'Connection' status of 'Connected' and buttons for 'Connect' and 'Disconnect'.
- Shutter:** Includes 'Shutter Type' (Detector output), 'Andor Shutter Mode' (Auto), and 'External shutter' (High To Open). It displays 'Status: Det. Closed' and 'EPICS Closed', along with 'Open/Close' buttons and 'Delay' values for 'Open' (0.100) and 'Close' (0.050).
- Plugins:** Features a list of plugins with checkboxes: 'All', 'File', 'ROI', 'Stats', and 'Other'.
- Readout:** A table-like interface for sensor parameters:

	X	Y
Sensor Size	1024	1024
Binning	1	1
Region Start	0	0
Region Size	1024	1024
Reverse	No	No
Image Size	1024	1024
Image Size (bytes)	2097152	
Gain	0	0
ADC Speed	2.5 MHz	2.5 MHz
Data type	UInt16	UInt16
- Collect:** Shows acquisition parameters: 'Exposure Time' (0.300), 'Accumulate Period' (0.000), 'Acquire Period' (0.100), '# Accums/Image' (1), '# Exposures Complete' (34), and '# Images/Acquis.' (2). It includes 'Image Mode' (Continuous) and 'Trigger Mode' (Internal) dropdowns, and 'Acquire' 'Start' and 'Stop' buttons. The 'Detector State' is 'Acquire' and 'Detector Status' is 'Data acquisition in progress'.
- Cooler:** Displays 'Cooler' (On), 'Temperature' (10.000), and 'Status' (Stabilized at set point).
- File:** Contains a 'Driver File I/O' checkbox.
- Attributes:** Includes a 'File' text field and a 'PAL File' section with a 'Path' field containing 'GREY.PAL'.

Perkin Elmer Flat Panel Driver

PerkinElmer.adl

Perkin Elmer Control - 13PE1:cam1:

Setup

asyn port PEDET1
EPICS name 13PE1:cam1:
Manufacturer Perkin Elmer
Model XRD0820
Connected
Connection
Debugging

Plugins

Readout

	X	Y
Sensor size	2048	2048
Binning	1	1
Image size	2048	2048
Image size (bytes)	8388608	

Setup

Frame Buffers 10 10
Frame buffer index 7
Image Number 31

Corrections

Corrections Directory

C:\Perkin_Elmer\
 Done
Correction Available

Offset

Offset Frames 10 10
 Done
Correction Available

Gain

Gain Frames 20 0
 Done
Correction Not Available

Bad Pixel File

16#5149_1pF_PxlMask.his
Correction Available

Shutter

Shutter mode None
Status: Det. Closed EPICS Closed
Open/Close
Delay: Open 0.000 Close 0.000
EPICS shutter setup

Collect

Exposure time 0.200 0.200
Gain 0.25pF 0.25pF
Images 20 20
Images complete 31
Skip frames Disable
Frames to skip 1 1
Image mode Continuous
Trigger mode Internal

Acquire
Detector state Collecting
Image counter 0 151
Image rate 5.0
Array callbacks Enable

Attributes

File PerkinElmerAttributes.xml

R2-0: Point Grey driver

- New driver for all cameras from Point Grey using their FlyCap2 SDK.
- Firewire, GigE and USB 3.0
- High performance, low cost



Mono
Sensor



Point Grey GigE Camera BlackFly PGE-20E4C

- e2v EV76C570 CMOS sensor
- Global shutter
- 29 x 29 x 30 mm
- Power Over Ethernet
- 4.5 micron pixels
- 1600 x 1200 pixels, color (mono)
- 47 frames/s
- \$595
 - 5X cheaper than comparable Prosilica cameras we bought in the past



Point Grey USB-3.0 Camera Grasshopper3 GS3-U3-23S6M

- 1920 x 1200 global shutter CMOS
- Sony IMX174 1/1.2
- No smear • Distortion-free
- Dynamic range of 73 dB
- Peak QE of 76%
- Read noise of 7e-
- 12-bit or 8-bit data
- Max frame rate of 162 fps
 - ~356 MB/S, >3X faster than GigE
- USB 3.0 interface
- \$1,295



Point Grey Driver

pointGrey.adl

Point Grey Area Detector Control - 13PG1:cam1:

Setup

asyn port PG1
 EPICS name 13PG1:cam1
 Manufacturer Point Grey Research
 Model Blackfly BFLY-PGE-20
 Serial Number 13481965
 Firmware Vers. 1.27.3.0
 Software Vers. 2.6.3

Debugging

Shutter

Shutter mode
 Status: Det. Closed EPICS Closed
 Open/Close
 Delay: Open Close
 EPICS shutter setup

Status

Status rate
 Dropped frames 0
 Corrupt frames 0
 Driver dropped 0
 Transmit failed 0
 Temperature 42.8

Attributes

File

Trigger

Internal
 Trigger mode
 Trigger source
 Trigger polarity
 Trigger delay
 Skip frames
 Software trigger

Strobe

Strobe source
 Strobe enable
 Strobe polarity
 Strobe delay
 Strobe duration

Bandwidth Control

Max packet size 9000
 Packet size
 Packet size 1440
 GigE packet delay
 Bandwidth (MB/s) 54.9

Plugins

All File ROI
 Stats Other

Readout

	X	Y
Sensor size	1600	1200
Region start	<input type="text" value="0"/>	<input type="text" value="0"/>
Region size	<input type="text" value="1600"/>	<input type="text" value="1200"/>
GigE binning	<input type="text" value="1x1"/>	<input type="text" value="1x1"/>
Image size	1600	1200
Image size (bytes)	1920000	
Gain	<input type="text" value="0.000"/>	<input type="text" value="0.000"/>
Data type	UInt8	
Color mode	Mono	
Video mode	<input type="text" value="Format7"/> <input type="text" value="Format7"/>	
Format7 mode	<input type="text" value="0 (1600x1200)"/>	
Properties	<input checked="" type="checkbox"/>	
Frame rate	<input checked="" type="checkbox"/> More <input type="text" value="Undefined1"/>	
Pixel format	<input checked="" type="checkbox"/> More <input type="text" value="Raw8"/>	
Convert raw	<input type="text" value="None"/> <input type="text" value="None"/>	
Timestamp	<input type="text" value="Camera"/> <input type="text" value="Camera"/>	

Collect

Exposure time
 Acquire period
 Frame rate
 # Images
 # Images complete 189
 # Exp./image
 Image mode

Collecting

Acquire
 Detector state Waiting
 Status

Image counter
 Image rate 30.0
 Array callbacks

Buffers

Buffers max/used	0	1
Buffers alloc/free	2	1
Memory max/used (MB)	0.0	3.7
Buffer & memory polling	<input type="text" value="1 second"/>	

Point Grey Driver (Grasshopper3 camera)

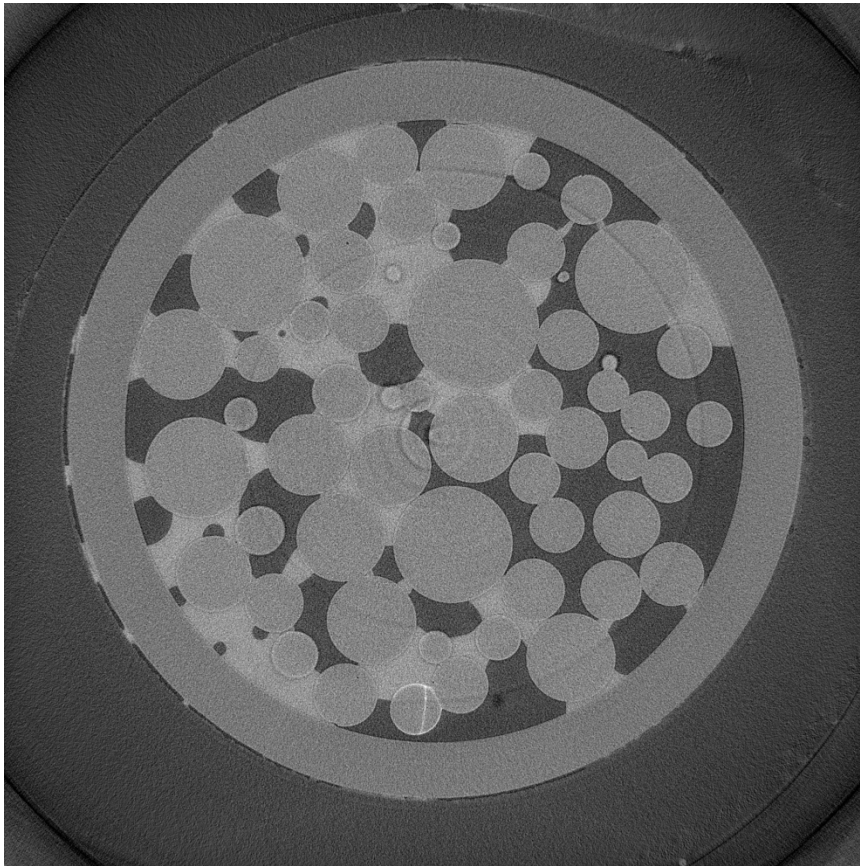
pointGreyProperties.adl

13PG1:cam1: Point Grey Properties

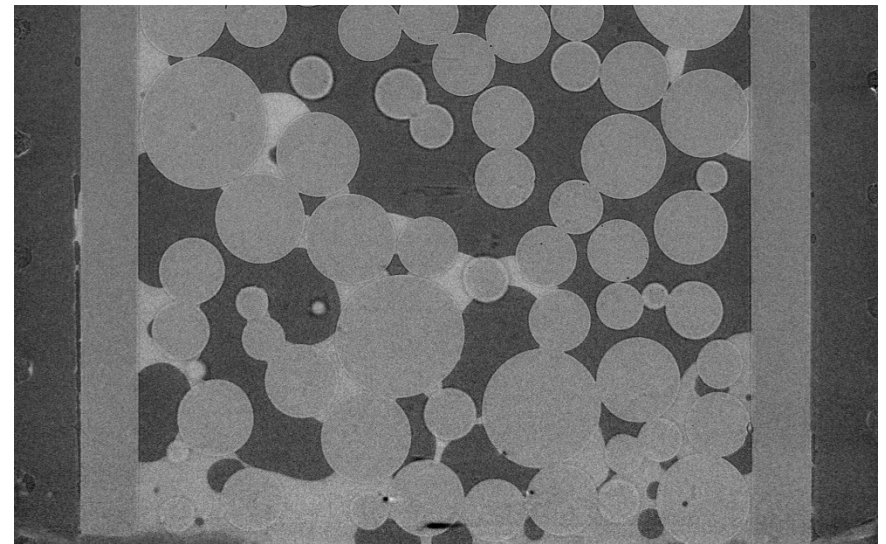
Property	Device Unit Control			Absolute Control			
	On/Off	One push	Auto/Manual	Set	Readback	Min	Max
Brightness	On			0	0	0	511
Auto exposure	On	Push	Manual	468	468	1	1023
Sharpness	Off		Manual	0	0	0	4095
White bal. red	Off						
White bal. blue							
Hue	Off						
Saturation	Off						
Gamma	On			512	512	512	4095
Shutter	On	Push	Manual	1181	1242	1	1242
Gain	On	Push	Manual	8	240	0	240
Iris	Off						
Focus	Off						
Temperature	Off						
Trigger mode	On		Manual	5	0	2844	1
Trigger delay	On			0	0	0	4095
Frame rate	Off		Manual	752	407	407	1629
Zoom	Off						
Pan	Off						
Tilt	Off						
				0.000	0.000	0.000	12.476
				-3.679	1.285	-7.585	2.414
				0.937	0.500	0.500	3.999
				23.779	76.311	0.061	76.311
				2.772	23.997	0.000	23.997
				20.000	3.133	0.000	inf
				0.000	0.000	0.000	0.077
				15.000	12.984	3.974	12.984

Pink Beam, Mirror=2.0 mrad

- Mirror angle=2.0 mrad (Beads_Pink_H)
- 2 mm Al absorber
- 8-bit data
- 1 ms exposure time, 124 frames/s, 900 projections, 7.3 seconds total
- Rotation axis orientation corrected for mirror angle



Horizontal slice



Vertical slice

Plugins

- Designed to perform real-time processing of data, running in the EPICS IOC (not over EPICS Channel Access)
- Receive NDAarray data over callbacks from drivers or other plugins
- Plug-ins can execute in their own threads (non-blocking) or in callback thread (blocking)
 - If non-blocking then NDAarray data is queued
 - Can drop images if queue is full
 - If executing in callback thread, no queuing, but slows device driver
- Allows
 - Enabling/disabling
 - Throttling rate (no more than 0.5 seconds, etc)
 - Changing data source for NDAarray callbacks to another driver or plugin
- Some plugins are also sources of NDAarray callbacks, as well as consumers.
 - Allows creating a data processing pipeline running at very high speed, each in a different thread, and hence in multiple cores on modern CPUs.

Plugins (continued)

- NDPlugInStdArrays
 - Receives arrays (images) from device drivers, converts to standard arrays, e.g. waveform records.
 - This plugin is what EPICS channel access viewers normally talk to.
- NDPluginROI
 - Performs region-of-interest calculations
 - Select a subregion. Optionally bin, reverse in either direction, convert data type.
 - Divide the array by a scale factor, which is useful for avoiding overflow when binning.
- NDPluginColorConvert
 - Convert from one color model to another (Mono, RGB1 (pixel), RGB2 (row) or RGB3 (planar) interleave)
 - Bayer conversion removed from this plugin, now part of Prosilica and Point Grey drivers.
- NDPluginTransform
 - Performs geometric operations (rotate, mirror in X or Y, etc.)

Plugins (continued)

- NDPluginStats
 - Calculates basic statistics on an array (min, max, sigma)
 - Optionally computes centroid position, width and tilt.
 - Optionally Computes X and Y profiles, including average profiles, profiles at the centroid position, and profiles at a user-defined cursor position.
 - Optionally computes the image histogram and entropy
- NDPluginROIStat
 - Multiple ROIs with simple statistics in a single plugin
 - More efficient when many ROIs are needed, e.g. for peaks in a 1-D energy spectrum
 - Min, max, total, net, mean
 - Time-series of each of these statistics

Plugins (continued)

- NDPluginProcess
 - Does arithmetic processing on arrays
 - Background subtraction.
 - Flat field normalization.
 - Offset and scale.
 - Low and high clipping.
 - Recursive filtering in the time domain.
 - Conversion to a different output data type.
- NDPluginOverlay
 - Adds graphic overlays to an image.
 - Can be used to display ROIs, multiple cursors, user-defined boxes, text, etc.
- ffmpegServer
 - MJPEG server that allows viewing images in a Web browser. From DLS.

Plugins (continued)

- **NDPluginAttribute**
 - Extracts NDAttributes from NDArrays and publishes their values as ai records
 - Can collect time-series arrays of the attribute values
- **NDPluginCircularBuff**
 - Buffers NDArrays in a circular buffer
 - Computes a trigger expression using up to 2 NDAttribute values
 - When trigger condition is met then outputs NDArrays
 - User-specified number of pre-trigger and post-trigger arrays to output
- **NDPluginTimeSeries**
 - Accepts 1-D NDArrays[NumSignals] or 2-D [NumSignals,NewTimePoints] and appends to time-series buffer
 - Operates in fixed length (stop when full) or circular buffer modes
 - Optional time-averaging of input data

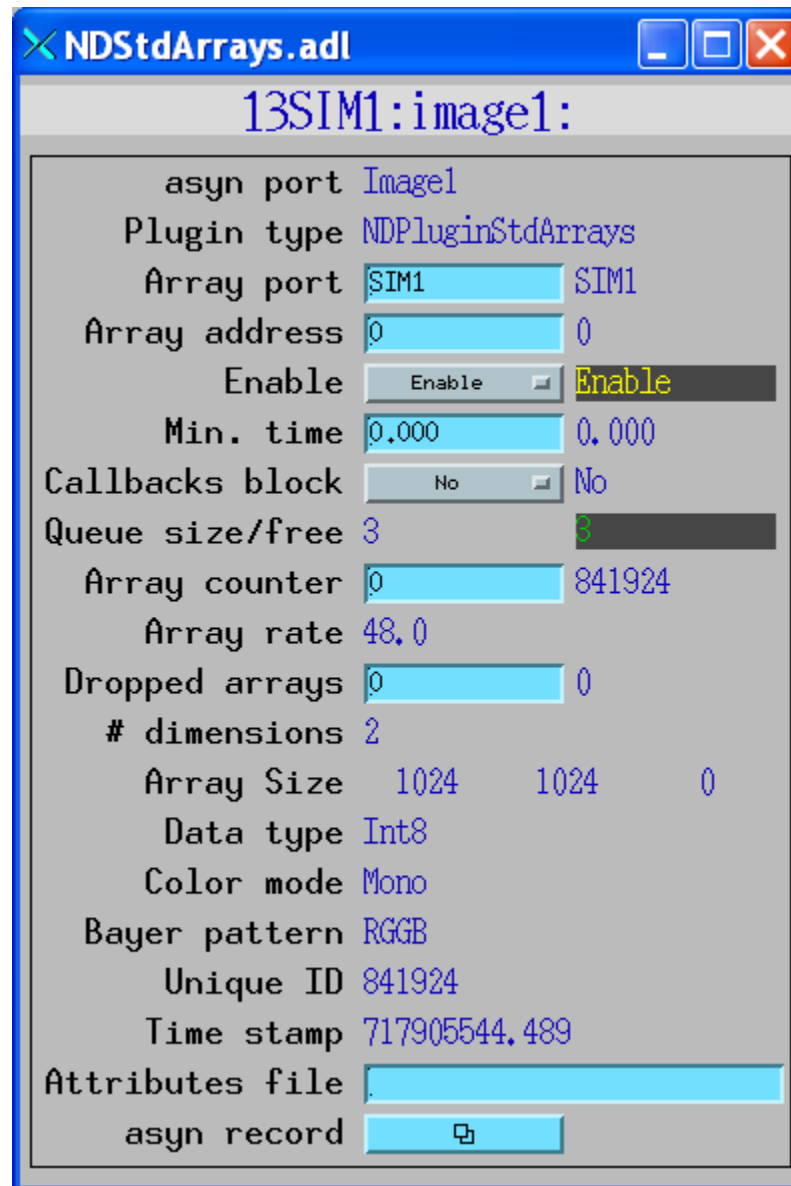
Plugins (continued)

- NDPluginFFT
 - Computes FFT of 1-D or 2-D NDArrays
 - Exports NDArrays containing the absolute value (power spectrum) of the FFT
 - Exports 1-D arrays of the FFT real, imaginary, absolute values, and time and frequency data.
- NDPluginPVA
 - Converts NDArrays to EPICS V4 NTNDArrays
 - Exports the NtNDArrays over PVAccess with internal V4 server
 - Can be used to send structured data to EPICS V4 clients
 - When used with the PVAccess driver then areaDetector plugins can be run on different machine from the detector driver

commonPlugins.adl All plugins at a glance

Plugin name	Plugin type	Port	Enable	Blocking	Dropped	Free	Rate	
Image1	NDPluginStdArrays	SIM1	Enable <input type="checkbox"/> Enable	No <input type="checkbox"/>	0	3	89.0	More
PROC1	NDPluginProcess	SIM1	Enable <input type="checkbox"/> Enable	No <input type="checkbox"/>	0	20	89.0	More
TRANS1	NDPluginTransform	SIM1	Disable <input type="checkbox"/> Disable	No <input type="checkbox"/>	0	20	0.0	More
CC1	NDPluginColorConvert	SIM1	Disable <input type="checkbox"/> Disable	No <input type="checkbox"/>	0	20	0.0	More
CC2	NDPluginColorConvert	SIM1	Disable <input type="checkbox"/> Disable	No <input type="checkbox"/>	0	20	0.0	More
OVER1	NDPluginOverlay	SIM1	Disable <input type="checkbox"/> Disable	No <input type="checkbox"/>	0	20	0.0	More
ROI1	NDPluginROI	SIM1	Enable <input type="checkbox"/> Enable	No <input type="checkbox"/>	0	19	89.0	More
ROI2	NDPluginROI	SIM1	Disable <input type="checkbox"/> Disable	No <input type="checkbox"/>	0	20	0.0	More
ROI3	NDPluginROI	SIM1	Disable <input type="checkbox"/> Disable	No <input type="checkbox"/>	0	20	0.0	More
ROI4	NDPluginROI	SIM1	Disable <input type="checkbox"/> Disable	No <input type="checkbox"/>	0	20	0.0	More
STATS1	NDPluginStats	ROI1	Disable <input type="checkbox"/> Disable	No <input type="checkbox"/>	0	20	0.0	More
STATS2	NDPluginStats	ROI2	Disable <input type="checkbox"/> Disable	No <input type="checkbox"/>	0	20	0.0	More
STATS3	NDPluginStats	ROI3	Disable <input type="checkbox"/> Disable	No <input type="checkbox"/>	0	20	0.0	More
STATS4	NDPluginStats	ROI4	Disable <input type="checkbox"/> Disable	No <input type="checkbox"/>	0	20	0.0	More
STATS5	NDPluginStats	SIM1	Enable <input type="checkbox"/> Enable	No <input type="checkbox"/>	885	0	21.0	More
FileNetCDF1	NDFileNetCDF	SIM1	Enable <input type="checkbox"/> Enable	No <input type="checkbox"/>	0	20	0.0	More
FileTIFF1	NDFileTIFF	SIM1	Disable <input type="checkbox"/> Disable	No <input type="checkbox"/>	0	20	0.0	More
FileJPEG1	NDFileJPEG	SIM1	Disable <input type="checkbox"/> Disable	No <input type="checkbox"/>	0	20	0.0	More
FileNexus1	NDPluginFile	SIM1	Enable <input type="checkbox"/> Enable	No <input type="checkbox"/>	0	20	0.0	More
FileMagick1	NDFileMagick	SIM1	Disable <input type="checkbox"/> Disable	No <input type="checkbox"/>	0	20	0.0	More
FileHDF1	NDFileHDF5 ver1.8.7	SIM1	Enable <input type="checkbox"/> Enable	No <input type="checkbox"/>	0	20	0.0	More

NDStdArrays plugin



ROI plugin

NDROI.adl

13SIM1:ROI1:

asyn port	ROI1		
Plugin type	NDPluginROI		
Array port	SIM1	SIM1	
Array address	0	0	
Enable	Enable	Enable	
Min. time	0.000	0.000	
Callbacks block	No	No	
Queue size/free	20	20	
Array counter	0	834794	
Array rate	48.0		
Dropped arrays	0	0	
# dimensions	2		
Array Size	1024	1024	0
Data type	Int8		
Color mode	Mono		
Bayer pattern	RGGB		
Unique ID	834794		
Time stamp	717905394.895		
Attributes file	i		
asyn record			

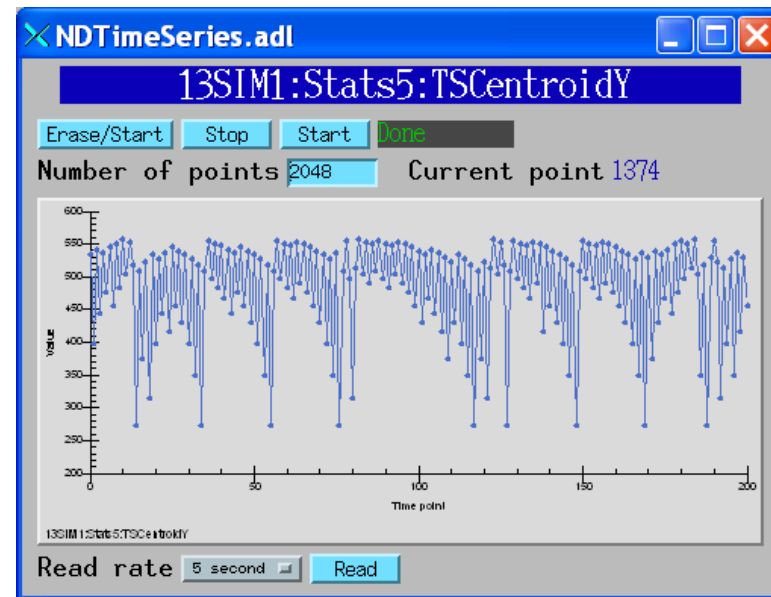
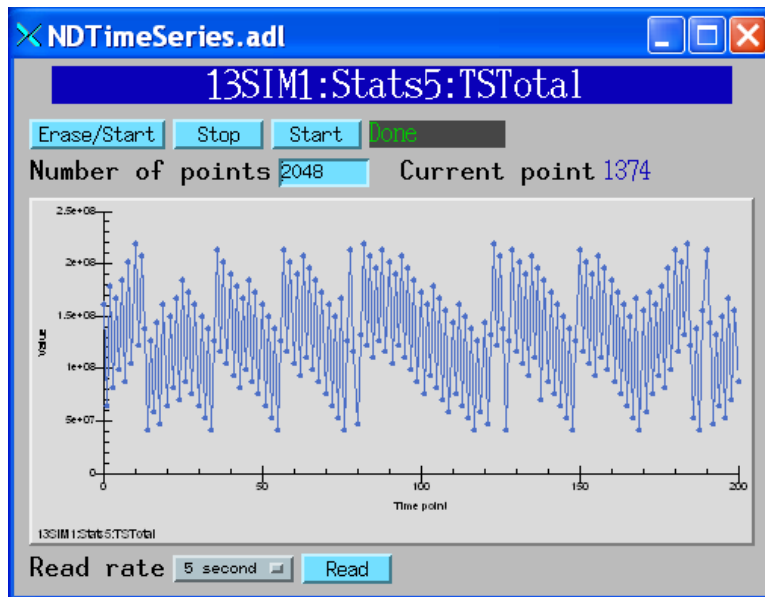
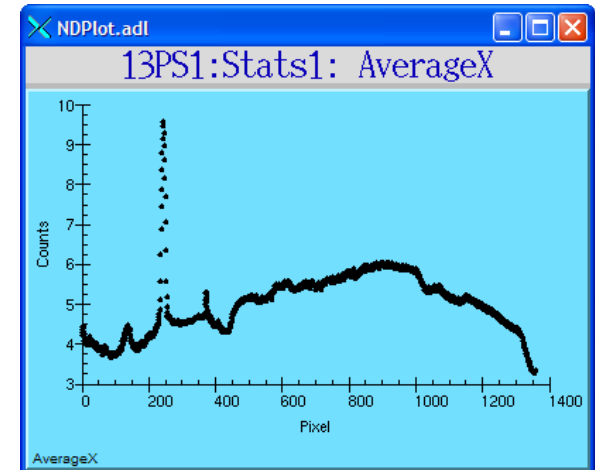
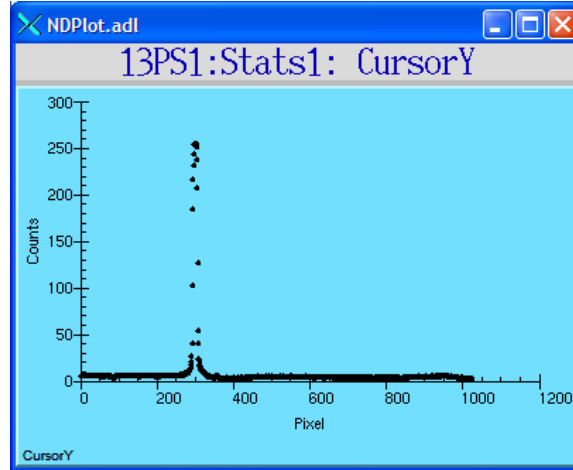
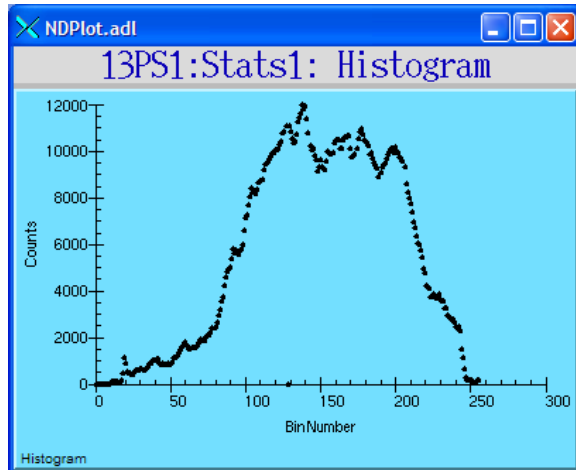
Definition			
Name	Upper left		
Data type	Automatic	Automatic	
Enable scaling	Enable	Enable	
Scale divisor	2	2	
	X	Y	Z
Input Size	1024	1024	0
Enable	Enable	Enable	Disable
	1	1	1
Binning	1	1	1
	0	0	0
ROI start	0	0	0
	512	512	1
ROI size	512	512	1
	No	No	No
Reverse	No	No	No
ROI Size	512	512	0

Statistics plugin

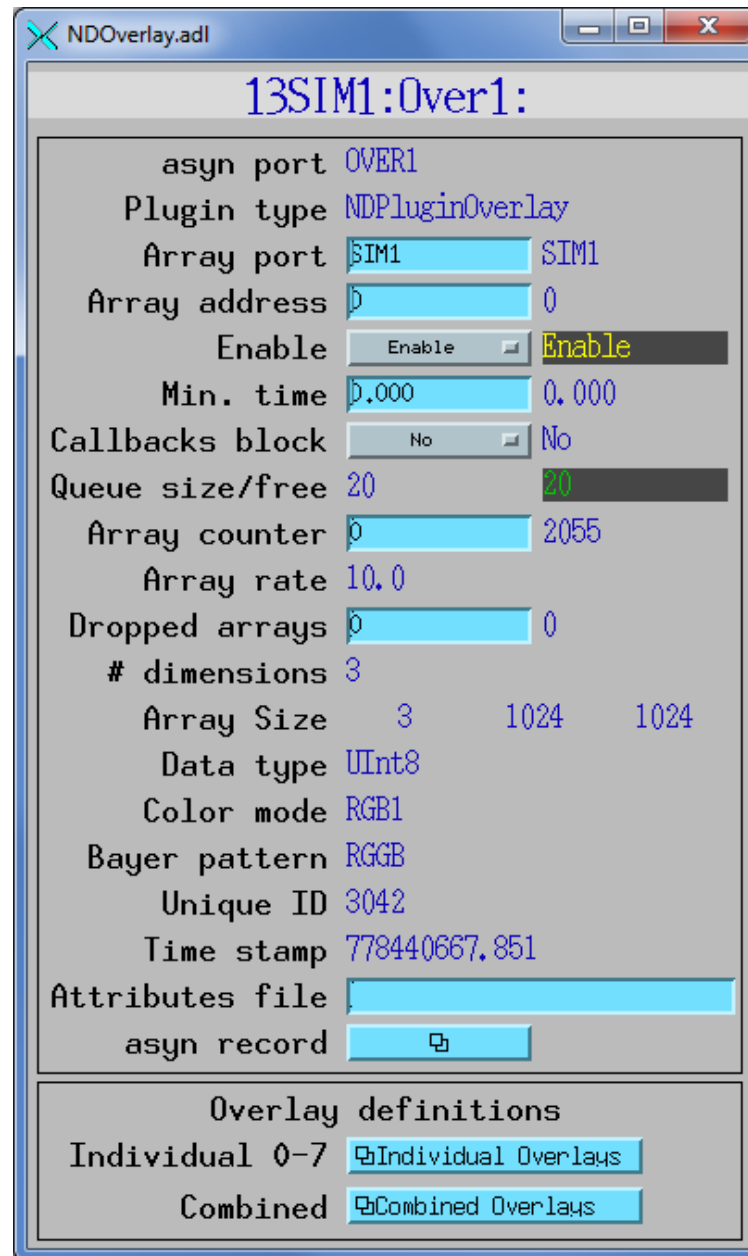
NStats.adl 13SIM1:Stats5:

<p>asyn port STATS5</p> <p>Plugin type NDPluginStats</p> <p>Array port SIM1 SIM1</p> <p>Array address 0 0</p> <p>Enable <input type="checkbox"/> Enable</p> <p>Min. time 0.000 0.000</p> <p>Callbacks block <input type="checkbox"/> No</p> <p>Queue size/free 20 12</p> <p>Array counter 0 4056</p> <p>Array rate 25.0</p> <p>Dropped arrays 0 395</p> <p># dimensions 2</p> <p>Array Size 1024 1024 0</p> <p>Data type Int8</p> <p>Color mode Mono</p> <p>Bayer pattern RGGB</p> <p>Unique ID 4451</p> <p>Time stamp 717886862.801</p> <p>Attributes file <input type="text"/></p> <p>asyn record <input type="checkbox"/></p>	<p>Statistics</p> <p>Compute statistics <input type="checkbox"/> Yes</p> <p>Background width 1 1</p> <p>Minimum 0 Maximum 6</p> <p>Min. X 0 Max. X 200</p> <p>Min. Y 0 Max. Y 148</p> <p>Total 622 Net 622</p> <p>Mean 0 Sigma 0.0</p> <p>Time series plots <input type="checkbox"/></p>	<p>Profiles</p> <p>Compute profiles <input type="checkbox"/> Yes</p> <p>Size X 1024 Y 1024</p> <p>256 256</p> <p>Cursor X <input type="text"/></p> <p>256 256</p> <p>Cursor Y <input type="text"/></p> <p>Plot <input type="checkbox"/></p>
	<p>Centroid</p> <p>Compute centroid <input type="checkbox"/> Yes</p> <p>Centroid threshold 1 1</p> <p>Centroid X 200.0 Y 150.1</p> <p>Sigma X 3.9 Y 3.9</p> <p>Sigma XY -0.024</p> <p>Time series plots <input type="checkbox"/></p>	<p>Histogram</p> <p>Compute histogram? <input type="checkbox"/> Yes</p> <p>Size 256 256</p> <p>Minimum 0 0</p> <p>Maximum 255 255</p> <p>Entropy -13.860</p> <p>Plot <input type="checkbox"/></p>
	<p>Time Series</p> <p>Erase/Start Stop Start Acquiring</p> <p>Number of points 2048</p> <p>Current point 82</p> <p>Read rate 5 second Read</p>	

Statistics plugin (continued)



Overlay plugin



Overlay plugin

NDOverlay8.adl

13PS1:Over1:

	Use?	Name	Shape	Draw mode	Red	Green (mono)	Blue	X Pos.	X Size	Y Pos.	Y Size	
	Yes	ROI1	Rectangle	Set	0	255	0	150	100	250	100	
1	<input checked="" type="checkbox"/>	ROI1	Rectangle	Set	0	255	0	150	100	250	100	More
	No	ROI2	Rectangle	XOR	0	255	0	0	100	0	100	
2	<input type="checkbox"/>	ROI2	Rectangle	XOR	0	255	0	0	100	0	100	More
	No	ROI3	Rectangle	Set	0	255	0	0	0	0	0	
3	<input type="checkbox"/>	ROI3	Rectangle	Set	0	255	0	0	0	0	0	More
	No	ROI4	Rectangle	Set	0	255	0	0	0	0	0	
4	<input type="checkbox"/>	ROI4	Rectangle	Set	0	255	0	0	0	0	0	More
	Yes	Cursor1	Cross	Set	0	255	0	420	50	196	50	
5	<input checked="" type="checkbox"/>	Cursor1	Cross	Set	0	255	0	420	50	196	50	More
	No	Cursor2	Cross	XOR	0	255	0	227	199	171	128	
6	<input type="checkbox"/>	Cursor2	Cross	XOR	0	255	0	227	199	171	128	More
	Yes	Box1	Rectangle	XOR	0	255	0	400	50	400	50	
7	<input checked="" type="checkbox"/>	Box1	Rectangle	XOR	0	255	0	400	50	400	50	More
	No	Box2	Rectangle	XOR	0	255	0	142	368	107	128	
8	<input type="checkbox"/>	Box2	Rectangle	XOR	0	255	0	142	368	107	128	More

Overlay plugin

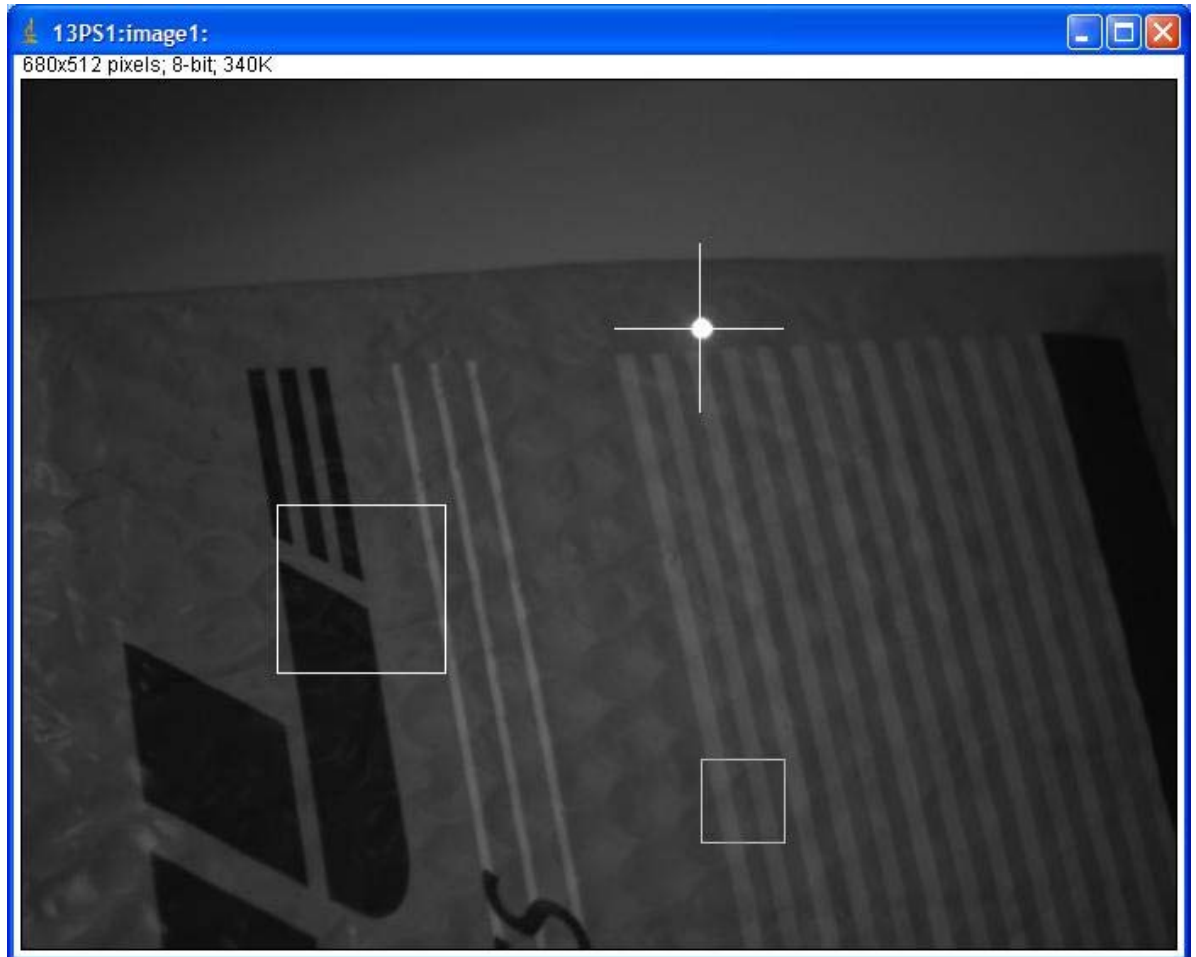
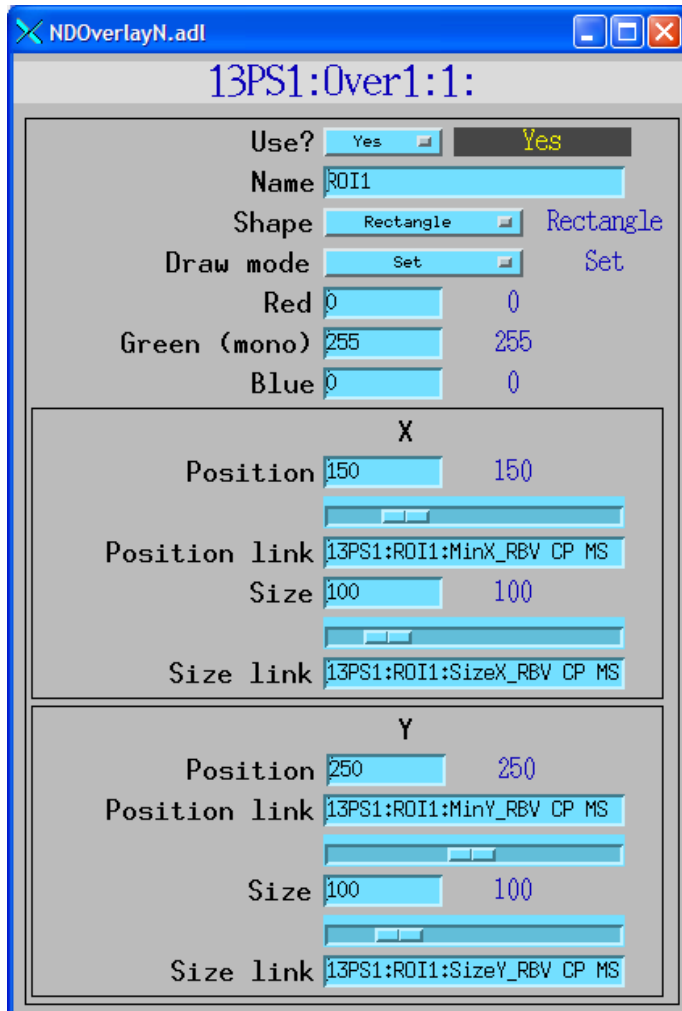
NDOverlayN.adl

13SIM1:Over1:1:

Use?	<input type="checkbox"/> Yes <input checked="" type="checkbox"/> Yes
Name	<input type="text" value="Text 1"/>
Shape	<input type="text" value="Text"/> Text
Draw mode	<input type="text" value="Set"/> Set
Red	<input type="text" value="255"/> 255
Green (mono)	<input type="text" value="255"/> 255
Blue	<input type="text" value="255"/> 255
Display Text	<input type="text" value="This is some text"/>
Time format	<input type="text" value="%Y-%m-%d %H:%M:%S.%03f"/>
Format example	<input type="text" value="%Y-%m-%d %H:%M:%S.%03f"/>
Font	<input type="text" value="6x13"/>

	X		Y	
Position	<input type="text" value="50"/> 50	<input type="text" value="50"/>	<input type="text" value="102"/> 102	<input type="text" value="102"/>
Position link	<input type="text" value="13SIM1:ROI1:MinX_RBV"/>		<input type="text" value="13SIM1:ROI1:MinY_RBV"/>	
Size	<input type="text" value="200"/> 200	<input type="text" value="200"/>	<input type="text" value="200"/> 200	<input type="text" value="200"/>
Size link	<input type="text" value="13SIM1:ROI1:SizeX_RBV"/>		<input type="text" value="13SIM1:ROI1:SizeY_RBV"/>	
Width	<input type="text" value="1"/> 1	<input type="text" value="1"/>	<input type="text" value="1"/> 1	<input type="text" value="1"/>
Width link	<input type="text" value="CP NPP MS"/>		<input type="text" value="CP NPP MS"/>	

Overlay plugin



Centroid of laser pointer calculated by statistics plugin

Cursor overlay X, Y position linked to centroid

Processing plugin

NDProcess.adl

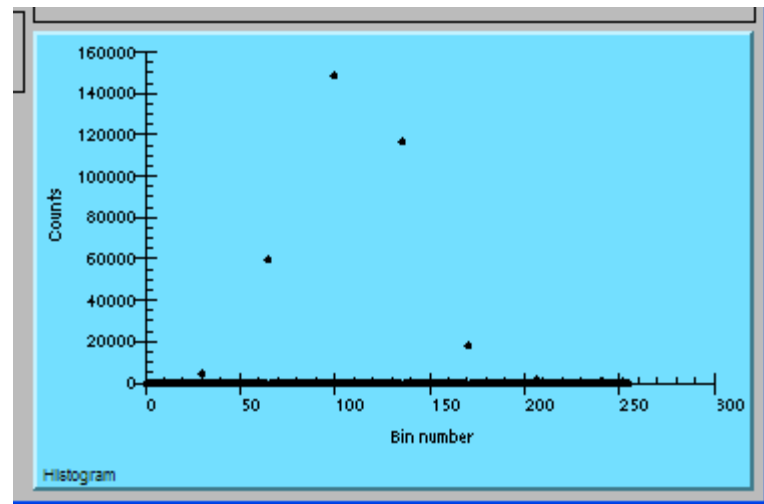
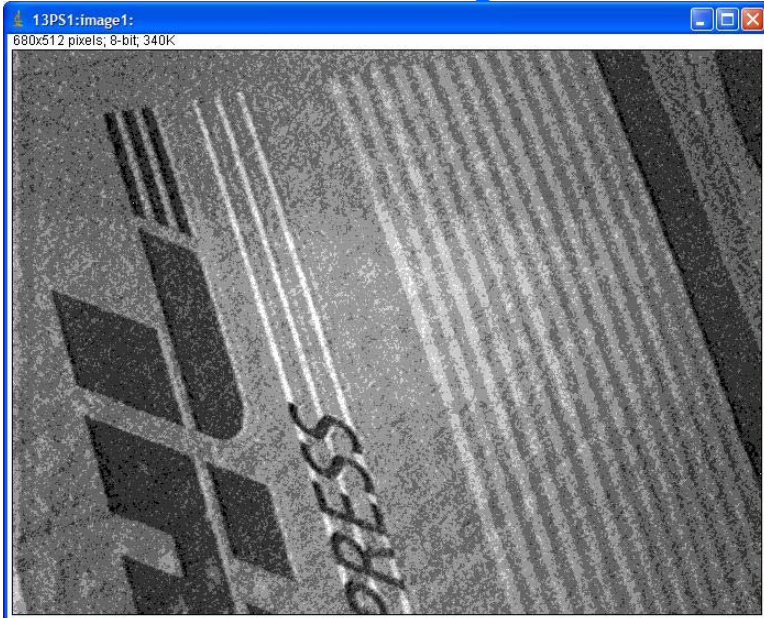
13SIM1:Proc1:

asyn port PROC1 Plugin type NDPluginProcess Array port SIM1 SIM1 Array address 0 0 Enable Enable Enable Min. time 0.000 0.000 Callbacks block No No Queue size/free 20 20 Array counter 0 11572 Array rate 47.0 Dropped arrays 0 0 # dimensions 2 Array Size 1024 1024 0 Data type Int8 Color mode Mono Bayer pattern RGGB Unique ID 12032 Time stamp 717887092.888 Attributes file asyn record	<h3>Background subtraction</h3> Save background Save Invalid Enable background Disable Disable <h3>Flat field normalization</h3> Save flat field Save Invalid Enable flat field Disable Disable Scale flat field 255 255 <h3>Scale and Offset</h3> Enable scale/off. Disable Enable Auto scale/off. Auto calc Scale value 0.10 42.50 Offset value 0.00 0.00 <h3>Low/High Clipping</h3> Enable low clip Disable Enable Low clip value 100 0 Enable high clip Disable Enable High clip value 150 255 <h3>Output data type</h3> Data type Automatic Automatic	<h3>Recursive filter</h3> Enable filter Disable Disable N filter 100 100 N filtered 0 Filter type RecursiveAve Reset filter Reset Auto reset filter Yes Filter callbacks Every array OOffset 0.00 0.00 OScale 1.00 1.00 OC1 1.00 1.00 OC2 -1.00 -1.00 OC3 0.00 0.00 OC4 1.00 1.00 FOffset 0.00 0.00 FScale 1.00 1.00 FC1 1.00 1.00 FC2 -1.00 -1.00 FC3 0.00 0.00 FC4 1.00 1.00 ROffset 0.00 0.00 RC1 0.00 0.00 RC2 1.00 1.00 $O[n] = Ooffset + OScale * ((OC1 + OC2/N) * F[n-1] + (OC3 + OC4/N) * I[n])$ $F[n] = Foffset + FScale * ((FC1 + FC2/N) * F[n-1] + (FC3 + FC4/N) * I[n])$ <p>On filter reset: $F[0] = ROffset + RC1 * F[n] + RC2 * I[0]$ I = Input array in callback F = Stored filter (double precision) N = value of NumFiltered O = Output array passed to clients</p>
--	---	---

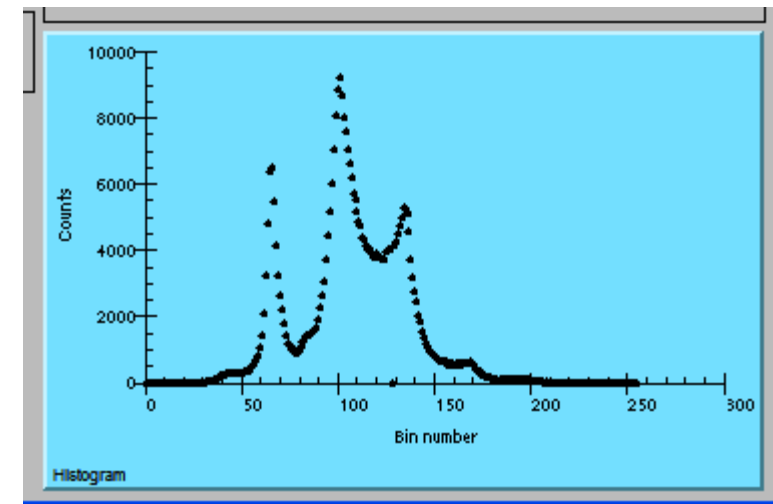
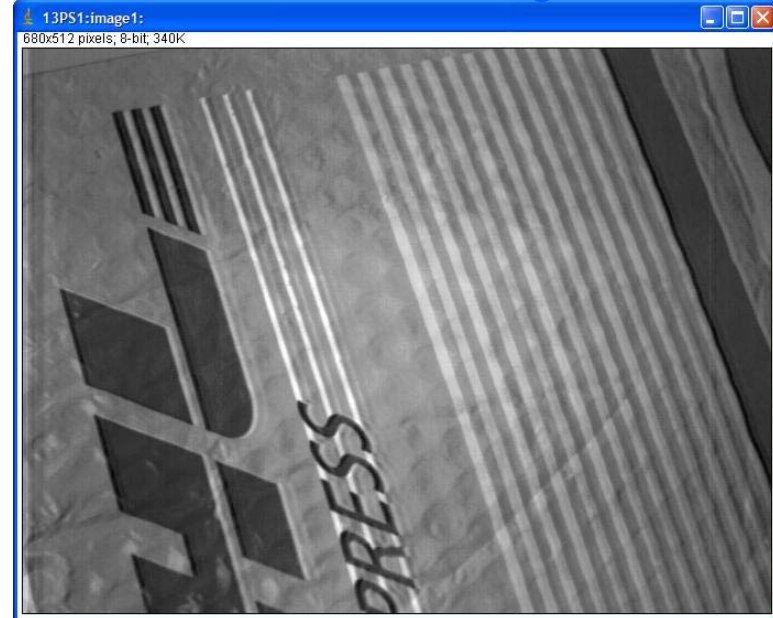
Processing plugin

30 microsec exposure time

No filtering



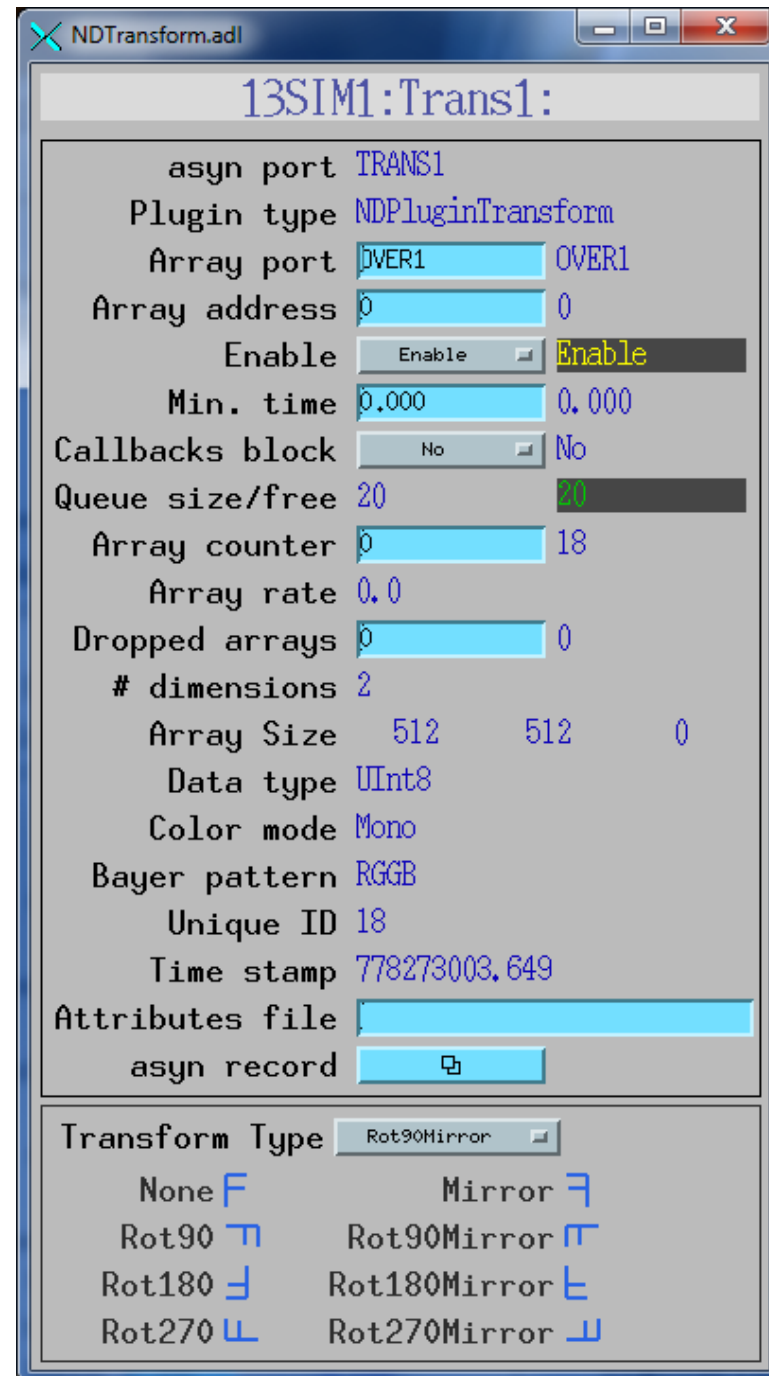
N=100 recursive average filter



Transform plugin

R2-1 changes

- Greatly simplified: just 8 operations including null operation
- 13-85 times faster than previous releases depending on data type, color mode



Plugins: NDPluginFile

- Saves NDArrays to disk
- 3 modes:
 - Single array per disk file
 - Capture N arrays in memory, write to disk either multiple files or as a single large file (for file formats that support this.)
 - Stream arrays to a single large disk file
- For file formats that support it, stores not just NDArray data but also NDAttributes

Plugins: NDPluginFile

- File formats currently supported
 - NDFileTIFF
 - Supports any NDArray data type
 - Stores NDAttributes as ASCII user tags
 - NDFileJPEG
 - With compression control
 - NDFileNetCDF
 - Popular self-describing binary format, supported by Unidata at UCAR
 - NDFileHDF5
 - Writes HDF5 files with the native HDF5 API, unlike the NeXus plugin which uses the NeXus API. Supports 3 types of compression.
 - Supports using an XML file to define the layout and placement of NDArrays and NDAttributes in the HDF5 file
 - R2-5 will support Single Writer Multiple Reader (SWMR). Only supported on local file systems, GPFS, and Lustre (not NFS or SMB)

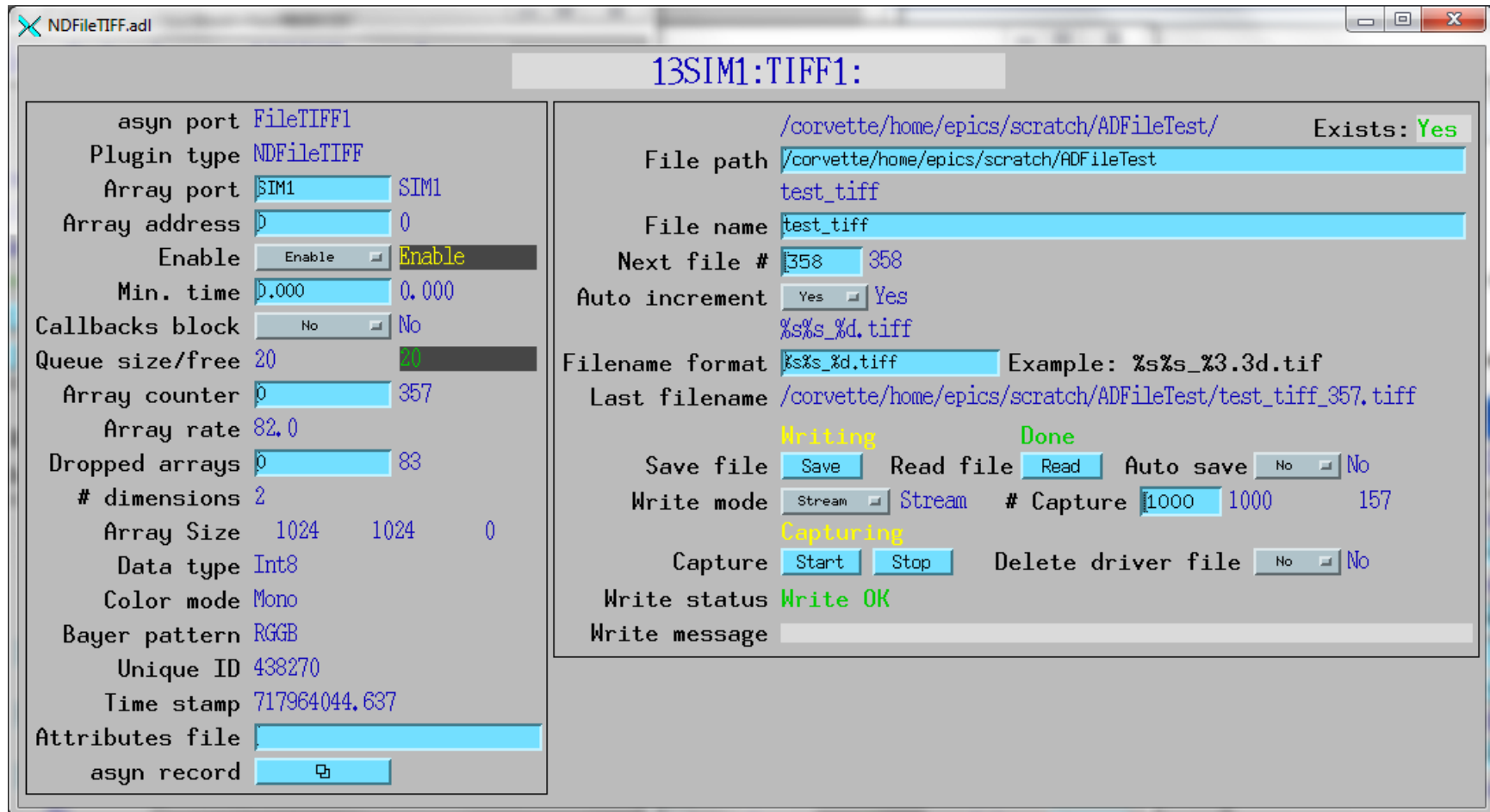
Plugins: NDPluginFile

- File formats currently supported
 - NDFileNeXus
 - Standard file format for neutron and x-ray communities, based on HDF5, which is another popular self-describing binary format; richer than netCDF
 - May be deprecated in a future release since NeXus files can now be produced with the NDFileHDF5 plugin using an appropriate XML layout file
 - NDFileMagick
 - Uses GraphicsMagick to write files, and can write in dozens of file formats, including JPEG, TIFF, PNG, PDF, etc.
 - NDFileNull
 - Used only to delete original driver files when no other file plugin is running

File saving with driver

- In addition to file saving plugins, many vendor libraries also support saving files (e.g. marCCD, mar345, Pilatus, etc.) and this is supported at the driver level.
- File saving plugin can be used instead of or in addition to vendor file saving
 - Can add additional metadata vendor does not support
 - Could write JPEGs for Web display every minute, etc.

NDPluginFile display: TIFF



Example: saving 82 frames/second of 1024x1024 video to TIFF files, a few dropped frames.

NDPluginFile display: netCDF

The screenshot shows the NDPluginFile netCDF interface for the device 13SIM1:netCDF1. The interface is divided into two main sections: configuration on the left and status/controls on the right.

Configuration (Left Panel):

- asyn port: FileNetCDF1
- Plugin type: NDFileNetCDF
- Array port: SIM1 (selected)
- Array address: 0
- Enable: Enable
- Min. time: 0.000
- Callbacks block: No
- Queue size/free: 20 / 20
- Array counter: 0 / 396
- Array rate: 47.0
- Dropped arrays: 0
- # dimensions: 2
- Array Size: 1024 x 1024 x 0
- Data type: Int8
- Color mode: Mono
- Bayer pattern: RGGB
- Unique ID: 1148948
- Time stamp: 717912009.118
- Attributes file: [empty field]
- asyn record:

Status and Controls (Right Panel):

- File path: /home/epics/scratch/ (Exists: Yes)
- File name: abc
- Next file #: 15
- Auto increment: Yes
- Filename format: %s%s_%d.nc (Example: %s%s_%3.3d.nc)
- Last filename: /home/epics/scratch/abc_14.nc
- Writing: Done
- Save file: Read file: Auto save: No
- Write mode: Stream Stream # Capture: 100 / 100 / 96
- Capturing: Delete driver file: No
- Write status: Write OK
- Write message: [empty field]

Example: streaming 47 frames/second of 1024x1024 video to netCDF files, no dropped frames.

NDFileHDF5

NDFileHDF5.adl

13SIM1:HDF1:

asyn port	FileHDF1
Plugin type	NDFileHDF5 ver1.8.7
Array port	SIM1 SIM1
Array address	0 0
Enable	<input type="checkbox"/> Enable <input checked="" type="checkbox"/> Enable
Min. time	0.000 0.000
Callbacks block	<input type="checkbox"/> No <input checked="" type="checkbox"/> No
Queue size/free	20 0
Array counter	0 611
Array rate	10.0
Dropped arrays	0 0
# dimensions	2
Array Size	1024 1024 0
Data type	UInt8
Color mode	Mono
Bayer pattern	RGGB
Unique ID	3461
Time stamp	779563295.068
Attributes file	
asyn record	<input type="checkbox"/>

File path	/home/epics/scratch/	Exists: Yes
File name	test_mono	
Next file #	220 220	
Auto increment	<input checked="" type="checkbox"/> Yes <input type="checkbox"/> Yes	
Filename format	%%s_%.3d.h5	Example: %%s_%.3d.h5
Last filename	/home/epics/scratch/test_mono_219.h5	
Lazy open	<input checked="" type="checkbox"/> Yes <input type="checkbox"/> Yes	
Save file	<input type="button" value="Save"/> <input type="button" value="Read file"/> <input type="button" value="Read"/> <input type="button" value="Auto save"/> <input type="checkbox"/> No <input checked="" type="checkbox"/> No	
Write mode	<input type="checkbox"/> Stream <input checked="" type="checkbox"/> Stream	# Capture 100 100 28
Capture	<input type="button" value="Start"/> <input type="button" value="Stop"/> <input type="button" value="Delete driver file"/> <input type="checkbox"/> No <input checked="" type="checkbox"/> No	
Write status	Write OK	
Write message		

Compression	<input type="checkbox"/> None <input checked="" type="checkbox"/> None	Extra dimensions
# data bits	8	# (0-2) 0 0
Data bits offset	0	Size N 1 1
SZip # pixels	16 16	Name N frame number n
Zlib level	6	Size X 1 1
Store performance	<input type="checkbox"/> No <input checked="" type="checkbox"/> Yes	Name X scan dimension X
Store attributes	<input type="checkbox"/> No <input checked="" type="checkbox"/> Yes	Size Y 1 1
Run time	9.913	Name Y scan dimension Y
I/O speed	80.7	

Exists: Yes

hdf5_layout_demo.xml

XML File name hdf5_layout_demo.xml

NDFileHDF5

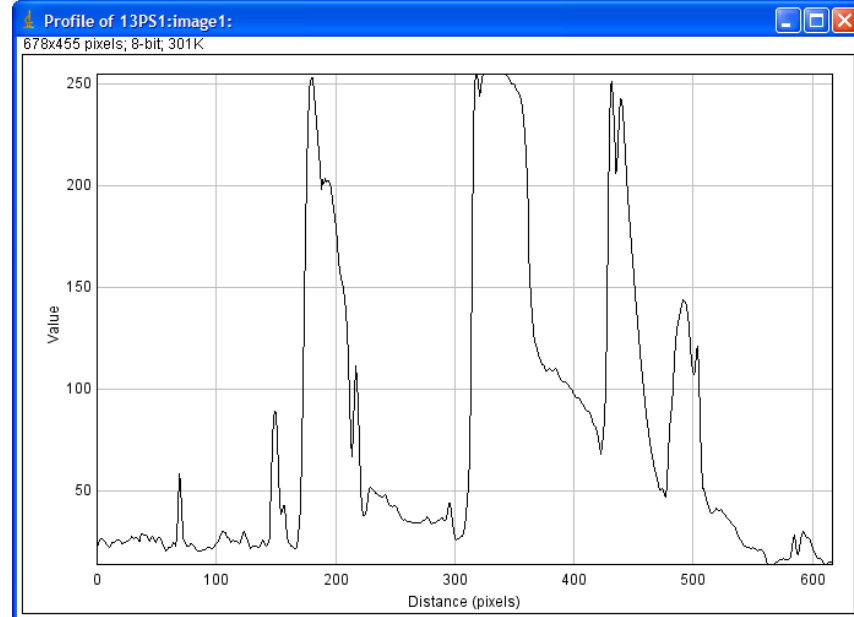
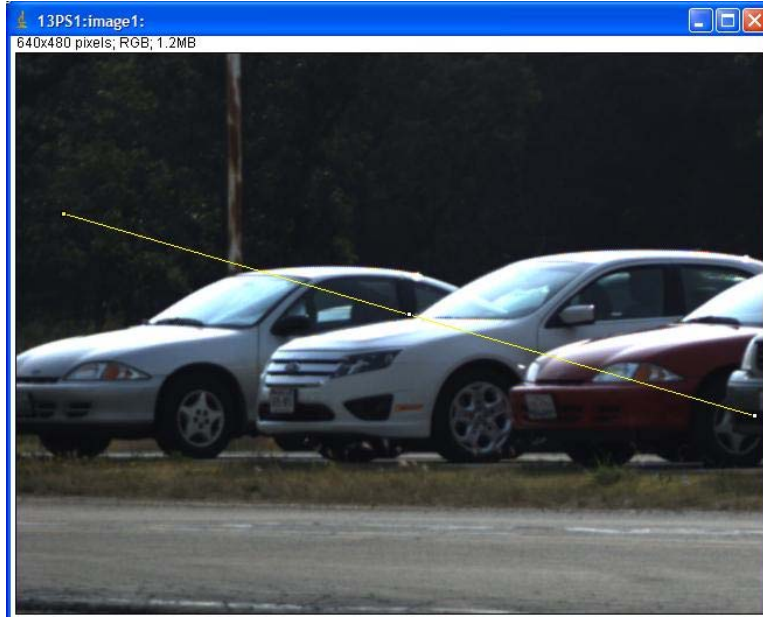
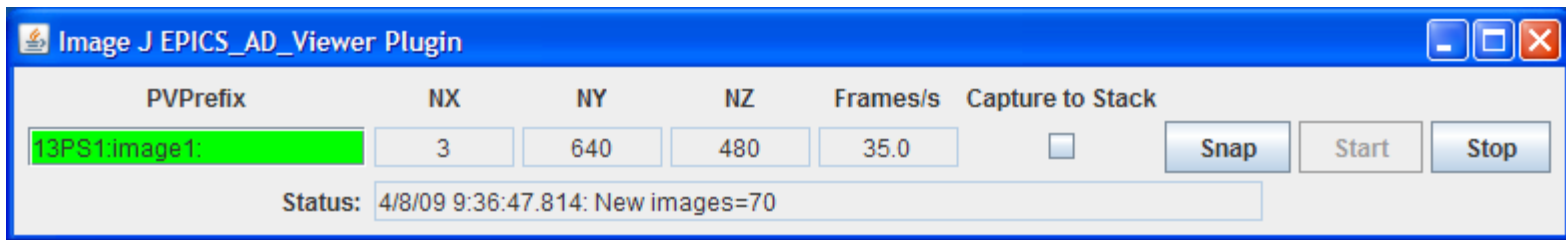
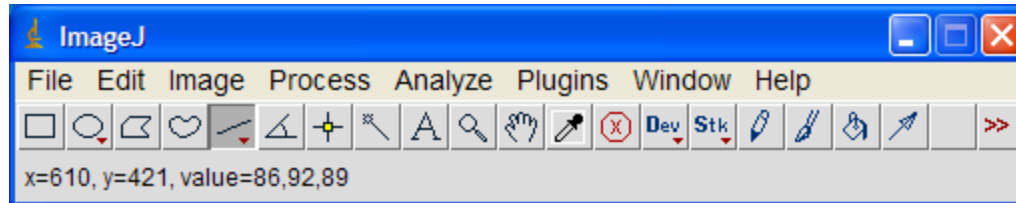
XML file to define file layout

```
<xml>
  <group name="entry">
    <attribute name="NX_class" source="constant" value="NXentry" type="string"></attribute>
    <group name="instrument">
      <attribute name="NX_class" source="constant" value="NXinstrument" type="string"></attribute>
      <group name="detector">
        <attribute name="NX_class" source="constant" value="NXdetector" type="string"></attribute>
        <dataset name="data" source="detector" det_default="true">
          <attribute name="NX_class" source="constant" value="SDS" type="string"></attribute>
          <attribute name="signal" source="constant" value="1" type="int"></attribute>
          <attribute name="target" source="constant" value="/entry/instrument/detector/data"
            type="string"></attribute>
        </dataset>
        <group name="NDAttributes">
          <attribute name="NX_class" source="constant" value="NXcollection" type="string"></attribute>
          <dataset name="ColorMode" source="ndattribute" ndattribute="ColorMode">
            </dataset>
          </group>
          <!-- end group NDAttribute -->
        </group>
        <!-- end group detector -->
      <group name="NDAttributes" ndattr_default="true">
        <attribute name="NX_class" source="constant" value="NXcollection" type="string"></attribute>
      </group>
      <!-- end group NDAttribute (default) -->
      <group name="performance">
        <dataset name="timestamp" source="ndattribute"></dataset>
      </group>
      <!-- end group performance -->
    </group>
    <!-- end group instrument -->
    <group name="data">
      <attribute name="NX_class" source="constant" value="NXdata" type="string"></attribute>
      <hardlink name="data" target="/entry/instrument/detector/data"></hardlink>
      <!-- The "target" attribute in /entry/instrument/detector/data is used to
        tell Nexus utilities that this is a hardlink -->
    </group>
    <!-- end group data -->
  </group>
  <!-- end group entry -->
</xml>
```

Viewers

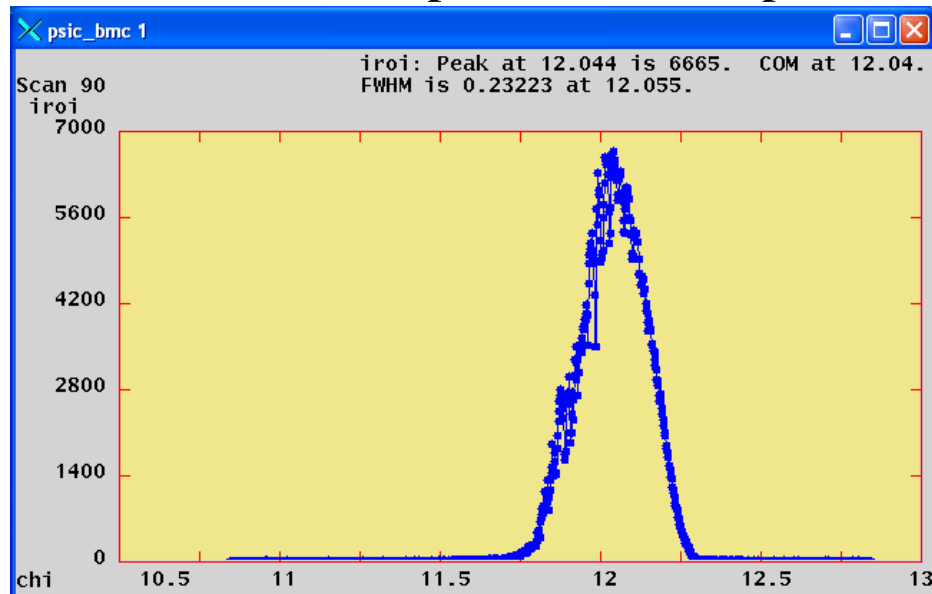
- areaDetector allows generic viewers to be written that receive images as EPICS waveform records over Channel Access
- Current viewers include:
 - ImageJ plugin EPICS_AD_Display. ImageJ is a very popular image analysis program, written in Java, derived from NIH Image.
 - IDL EPICS_AD_Display.
 - ffmpegServer allows image display in any Web browser
 - ffmpegViewer high-performance Qt-based viewer for MJPEG stream

ImageJ Viewer



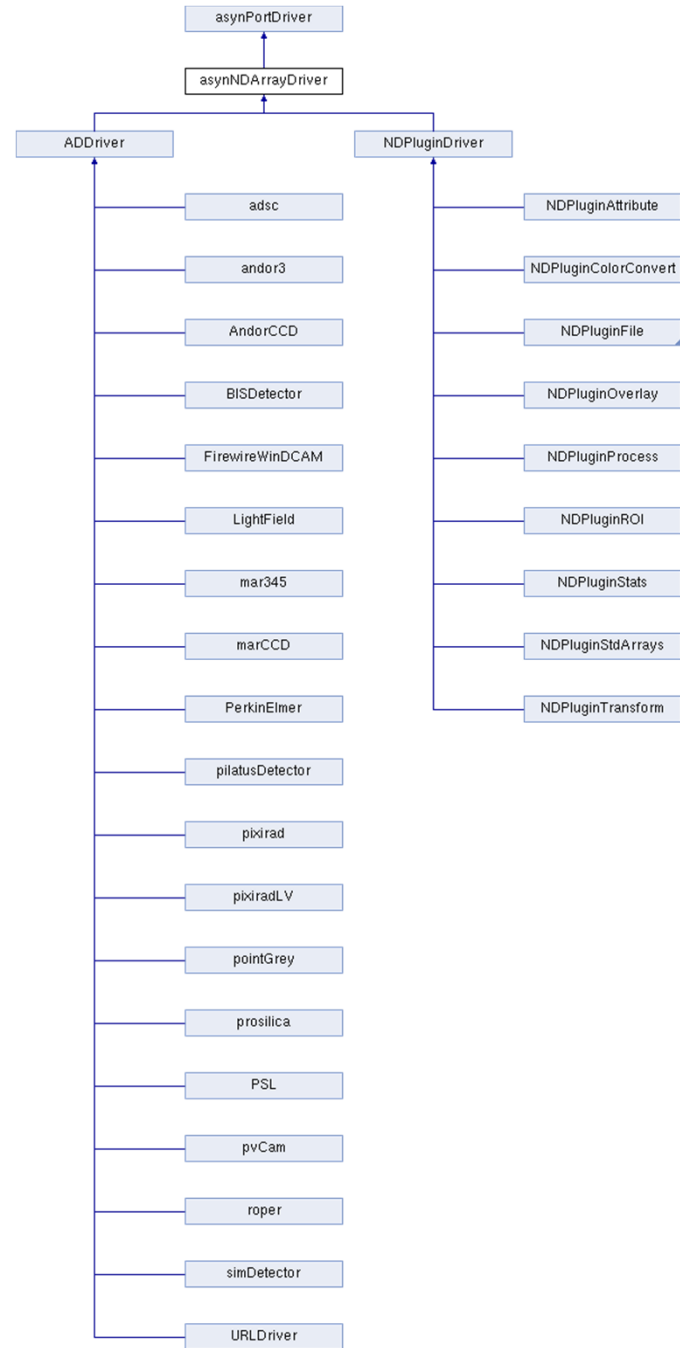
Performance Example with Pilatus driver

- SPEC used to collect 1000 points using trajectory scanning mode with the Newport XPS motor controller. Hardware trigger of Pilatus from XPS.
- Relative scan of the chi axis from -2 degrees to +2 degrees with 1000 points at .02 seconds/point
- Coordinated motion of the phi, kappa and omega axes.
- Theoretical time 20.0 second, actual time 20.8 seconds
- Includes time to save all 1000 images to disk (366 MB), Pilatus driver to read each file, correct bad pixels and flat field, compute ROIs, and post the ROIs and 1000 images to EPICS.



Internals

Class hierarchy



ADCore R3-0

- Simplify NDPluginFile base class and way file saving works
 - Remove the Single/Stream/Capture mode.
- Two parameters
 - # NDArrays to save (already present)
 - # NDArrays per file (new)
 - This allows saving only 1 array per HDF5 file, which is not possible now in Stream mode.
- Capture mode can be replaced:
 - Make input queue large enough OR
 - Use new NDPluginCircularBuffer
- Will require modifying clients that are doing file saving, hence a major release number

Future Ideas

- Put more functionality into ADDriver base class
 - Currently it does not do much, all code is in each driver for:
 - Doing callbacks to plugins
 - Processing new exposure time with writeFloat64 function
 - writeFloat64 in ADDriver base class would call setExposure() in derived class
 - Derived class would call ADDriver::doPluginCallbacks(), which would handle setting attributes, getting timestamp, calling plugins, etc.
- This is the way the Model 3 motor driver, which also uses asynPortDriver, is written
- Demultiplexor/multiplexor plugin
 - Allow multiple plugins to work on the same data stream when it saturates a single core

Conclusions

- Architecture works well, easily extended to new detector drivers, new plugins and new clients
- Base classes, `asynPortDriver`, `asynNDArrayDriver`, `asynPluginDriver` actually are generic, nothing “areaDetector” specific about them.
- They can be used to implement any N-dimension detector, e.g. the XIA xMAP (16 detectors x 2048 channels x 512 points in a scan line)
- Can get documentation and pre-built binaries (Linux, Windows, Cygwin) from our Web site:
 - <http://cars.uchicago.edu/software/epics/areaDetector>
- Can get code from github
 - <https://github.com/areaDetector>

Acknowledgments

- Brian Tieman, John Hammonds (APS) Perkin-Elmer driver, NeXus file saving plugin, Roper PVCAM driver
- Tim Madden (APS) initial version of ImageJ viewer, PCO drivers
- Ulrik Pedersen, Matt Pearson, Tom Cobb, David Hickin, Alan Greer (Diamond and Observatory Sciences) ffmpeg plugin, aravisGigE driver, Linux Firewire driver, HDF5 plugin
- Bruno Martins (BNL) V4 plugin and driver
- Lewis Muir (APS IMCA CAT) ADSC CCD driver
- Chris Roehrig (APS) new NDPluginTransform
- Many others for enhancements and bug fixes
- NSF-EAR and DOE-Geosciences for support of GSECARS where most of this work was done