OAK RIDGE National Laboratory

Area Detector

Oct. 2018

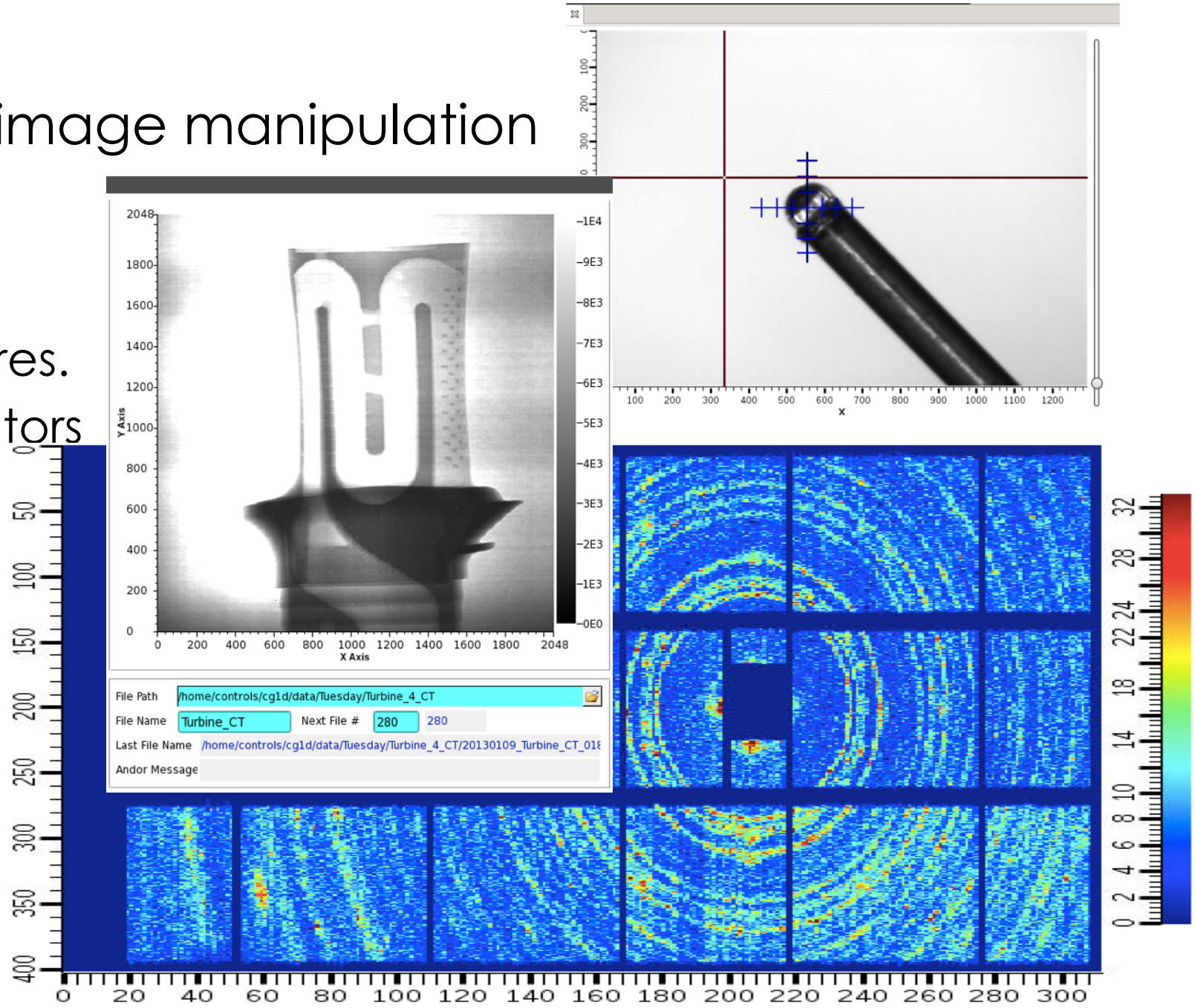
Kay Kasemir, Klemen Vodopivec
based on presentations by Mark Rivers, APS, U. Chicago

ORNL is managed by UT-Battelle, LLC for the US Department of Energy

U.S. DEPARTMENT OF ENERGY

Area Detector

- EPICS framework for image manipulation
- Cameras
  - Cheap “Web Cam”
  - $$$ high speed, high res.
  - Neutron, X-Ray detectors
- Plugins collection
  - ROI
  - Transform
  - ColorConvert
  - Etc.
- Extendibility



Features

- Maybe the largest shared EPICS Application
- PVs for image settings, shutter, exposure, …
  - "Simulated" area detector IOC has 6000 records
- N-D data
  - 1D time series data
  - 2D images (most plugins)
  - N<=10
  - Custom metadata
- Supports >500 frame/second detectors

Disclaimer

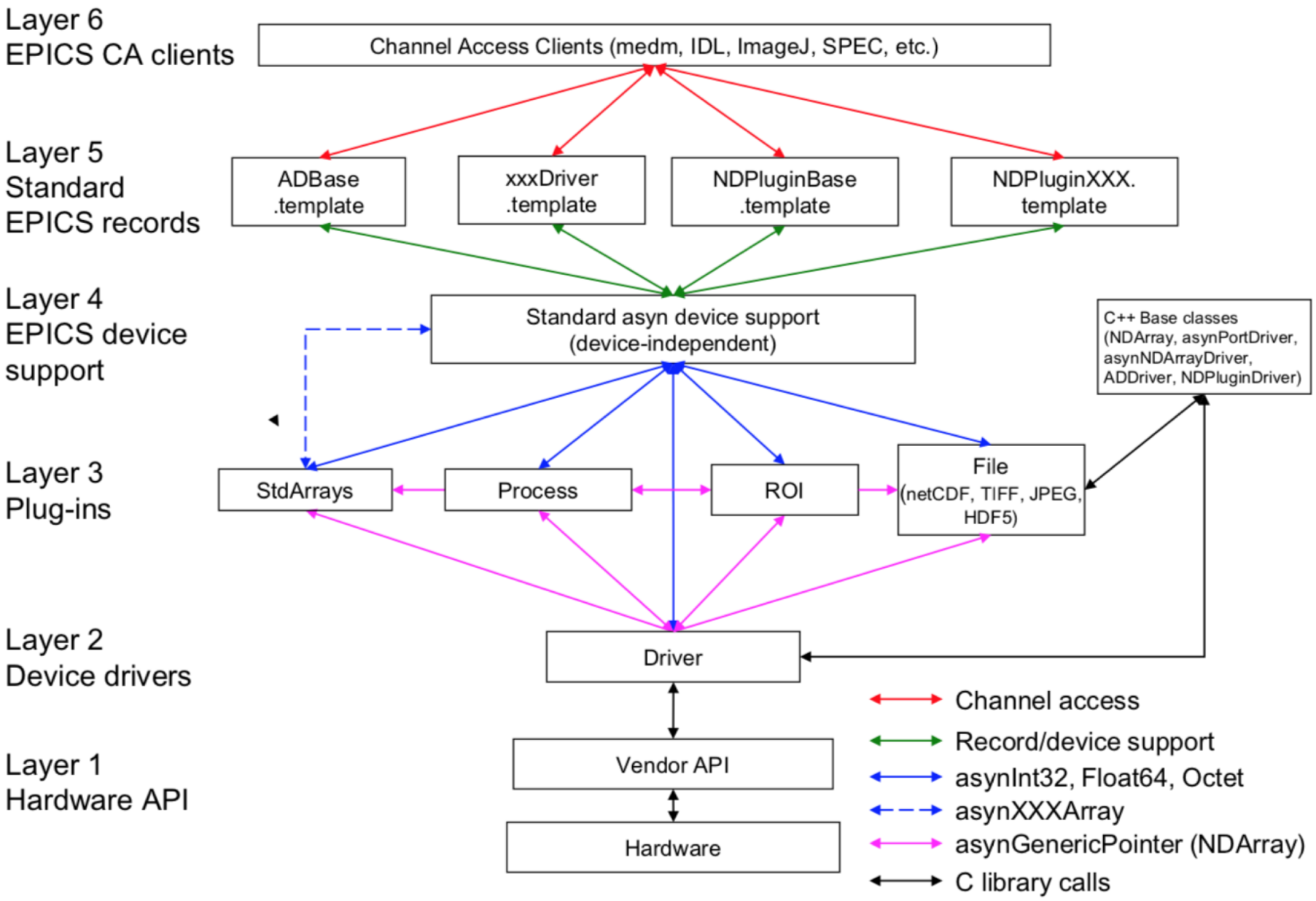
This will only scratch the surface.

EPICS web site has several days of training material if you are serious about using the A.D.

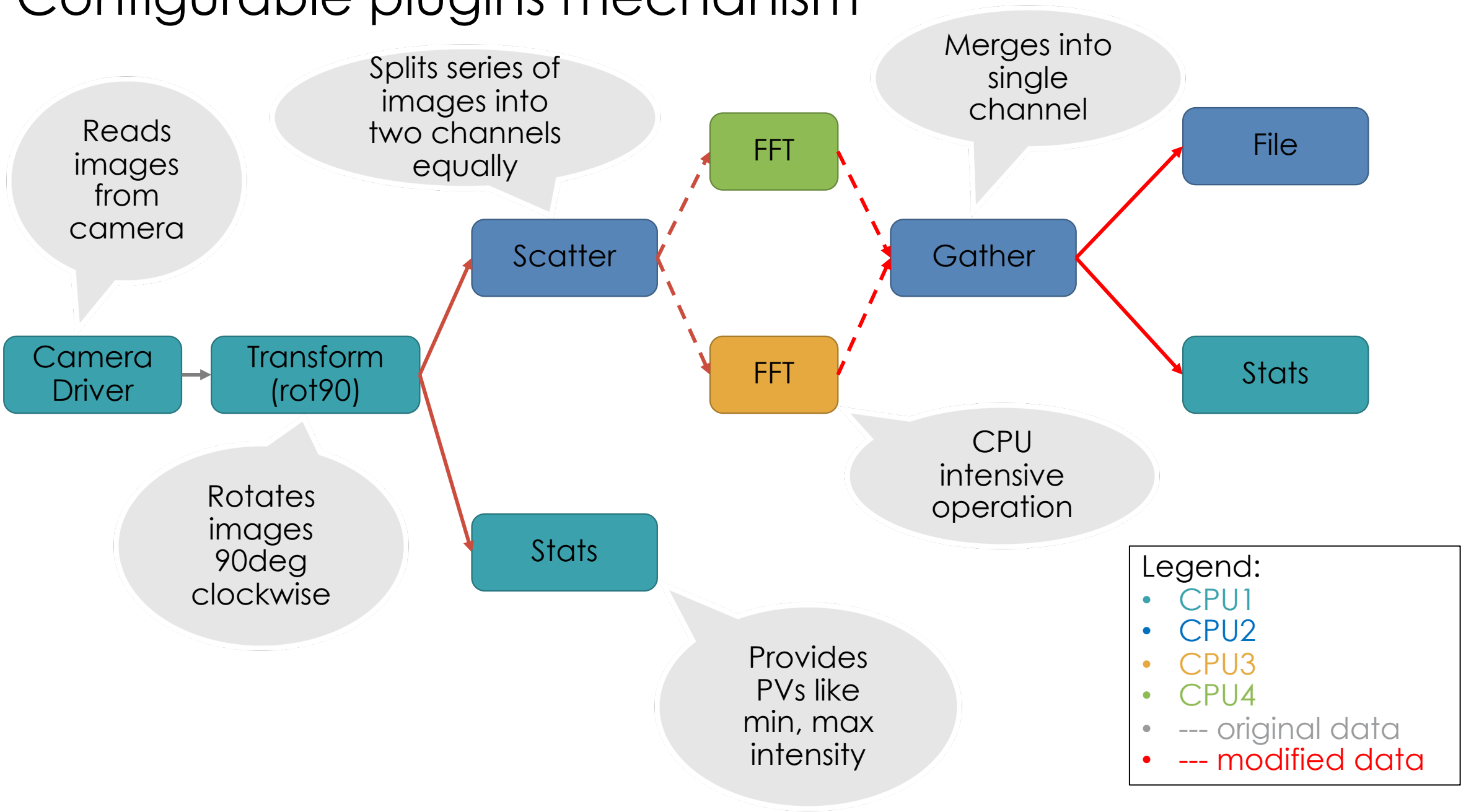
Vocabulary

- Driver
  - Interface to camera
  - Vendor libraries, custom protocols
  - Creates NDArray
- Pugin
  - Manipulates NDArray data
  - May change data
  - May send data to other plugins
    - No-copy if not changed
- NDArray
  - Structure holding data
  - N-dimensional array
    - N=2 for basic greyscale image
    - N>2 for color, detector with “depth”
- NDAttribute
  - Metadata attached to NDArray
    - Motor position, temperature, shutter,…
  - Added by driver, from PVs, Plugins…
- NDArrayPool
  - Pool of NDArrays to reduce memory allocation

EPICS AreaDetector Architecture



Configurable plugins mechanism

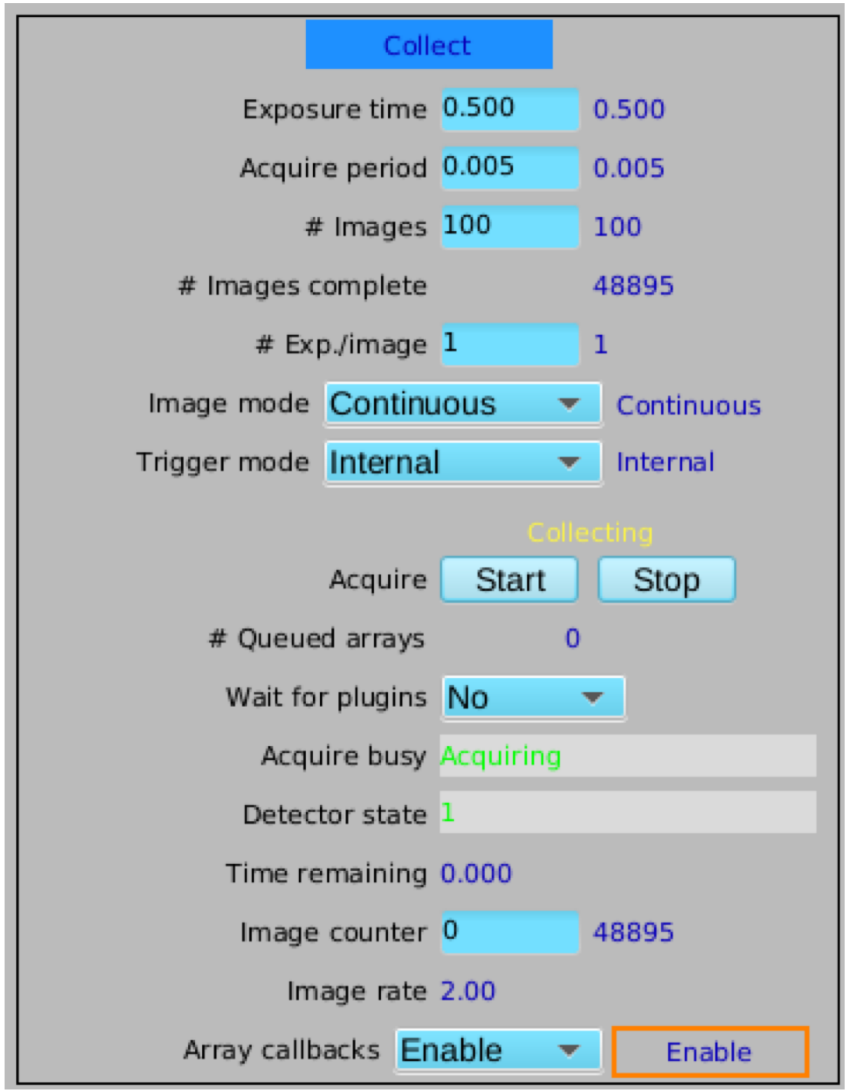


ADSimDetector

- Simulated images

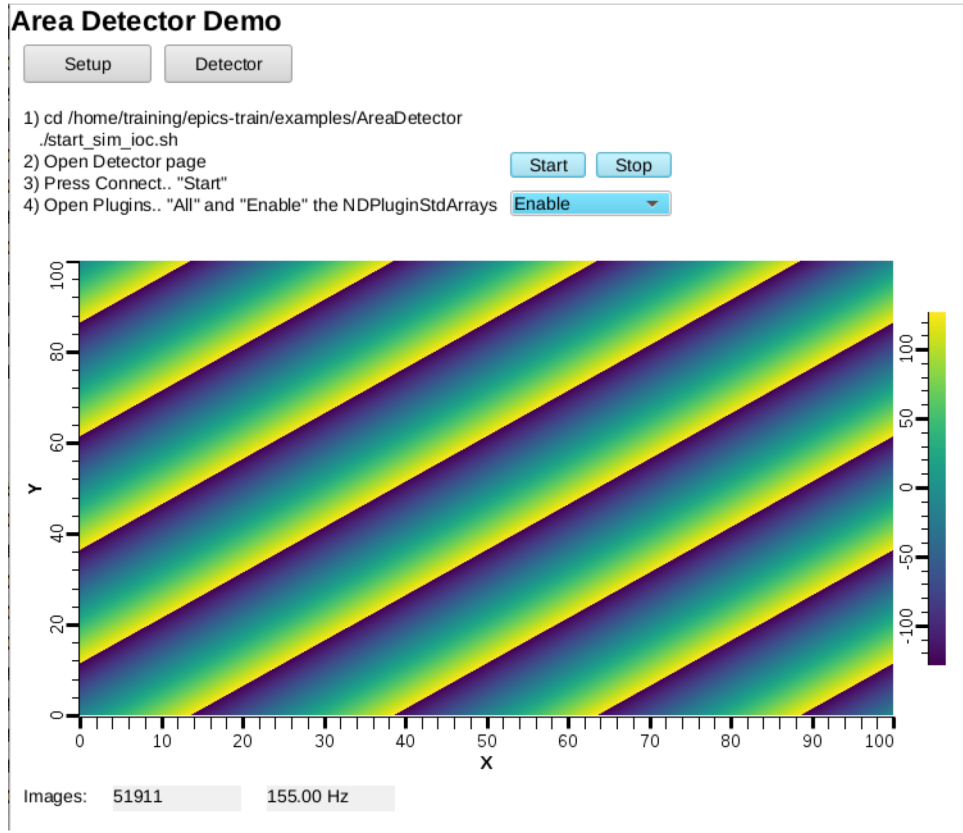
```
cd ~/epics-train/examples/AreaDetector
./start_sim_ioc.sh
```

- Open the AreaDetectorDemo.bob
  - On “Detector” page,
    “Start” the SIM1 detector



NDPluginStdArrays

- Serves image as Channel Access waveform
- On Detector, Plugins, All, find NDPluginStdArrays
  - Port = “SIM1”
  - Enable



- AreaDetectorDemo.bob shows image
  - PV: 13SIM1:image1:ArrayData
  - Width x Height: 1024 x 1024
  - Unsigned

NDPluginOverlay

- Adds rectangles, text etc. to image
- On Detector, Plugins, All, find NDPluginOverlay "OVER1"
  - Set its Port to "SIM1", Enable
  - Change NDPluginStdArrays's Port to "OVER1"

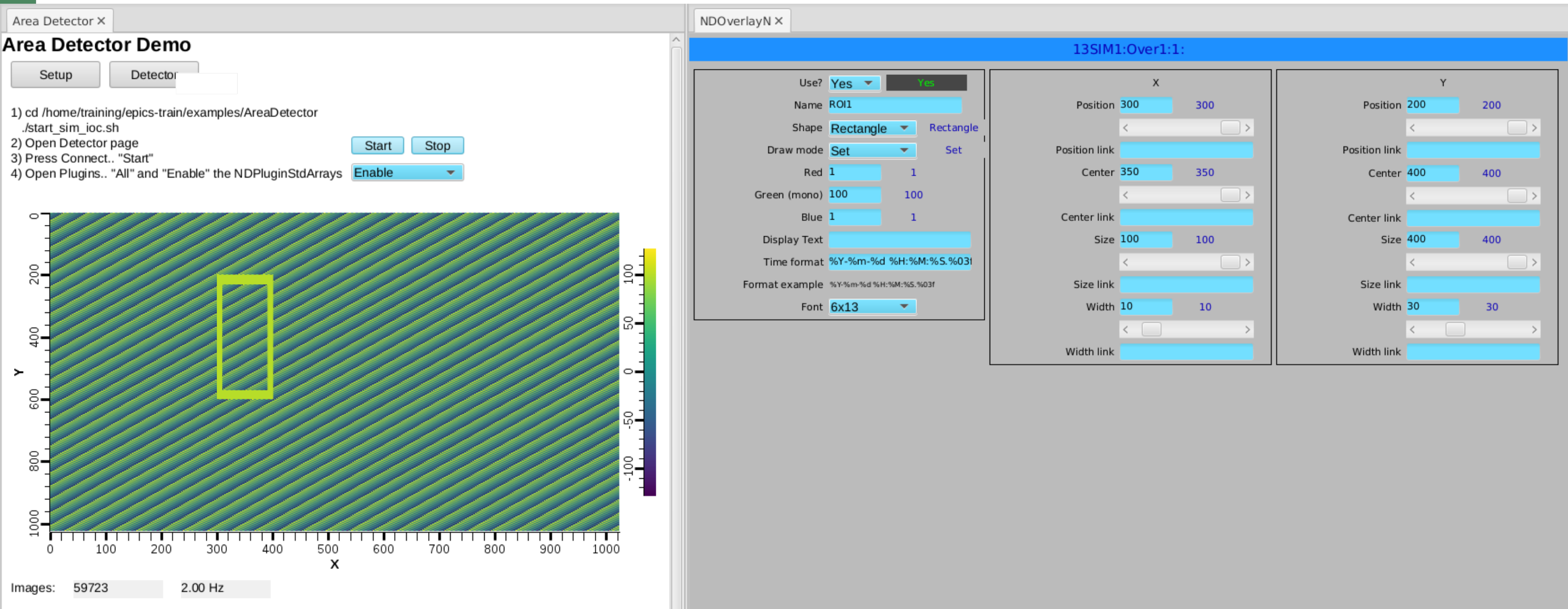
13SIM1: Common Plugins

Plugin name	Plugin type	Port	Enable		Blocking	Dropped	Free	Rate	
Image1	NDPluginStdArrays	OVER1	Enable	Enable	No	0	20	2.00	More
<13SIM1:Pva1:Port	<13SIM1:Pva1:PluginType_R	<13SIM1:P	<null>	<13SIM1:Pva	<null>	<13SIM1:Pva	<13SIM1:Pva	<13SIM1:Pva	More
PROC1	NDPluginProcess	SIM1	Disabl	Disable	No	0	20	0.00	More
TRANS1	NDPluginTransform	SIM1	Disabl	Disable	No	0	20	0.00	More
CC1	NDPluginColorConvert	SIM1	Disabl	Disable	No	0	20	0.00	More
CC2	NDPluginColorConvert	SIM1	Disabl	Disable	No	0	20	0.00	More
OVER1	NDPluginOverlay	SIM1	Enable	Enable	No	0	20	2.00	More

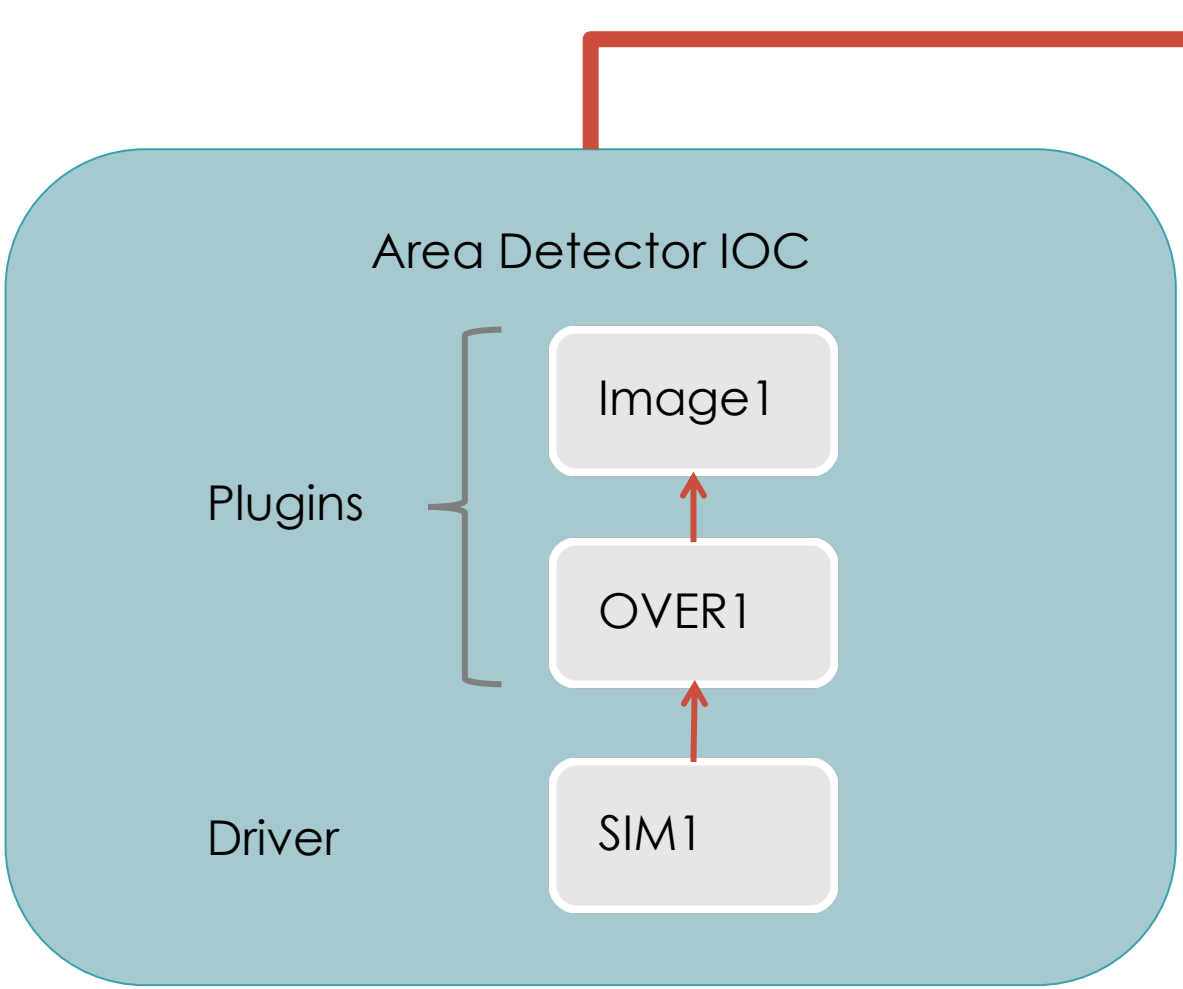
  - Press "More", select first of the "Individual Overlays"

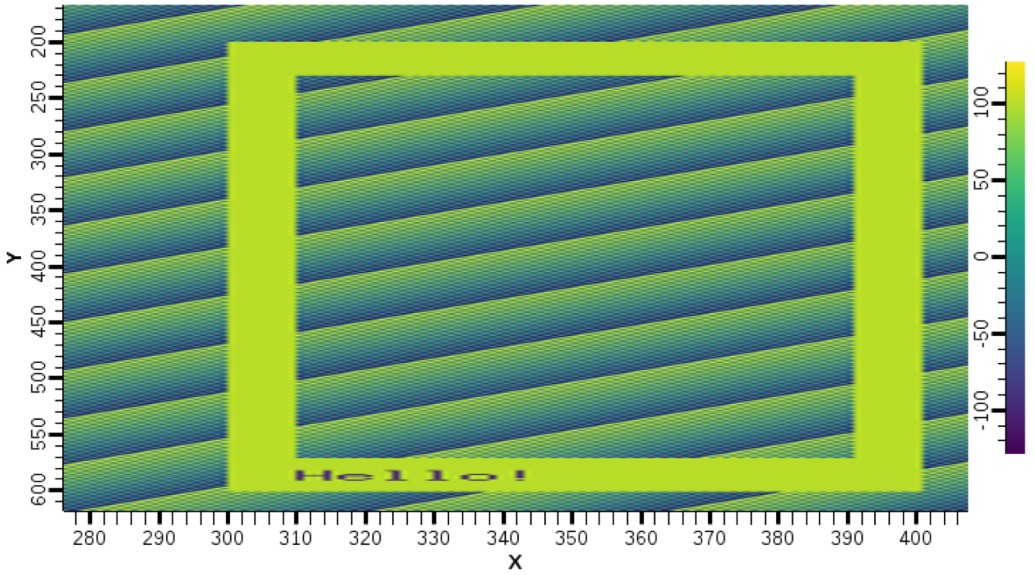
NDPluginOverlay.. Overlay #1

Set Use: Yes, Shape: Rectangle, set X and Y as shown



What we did



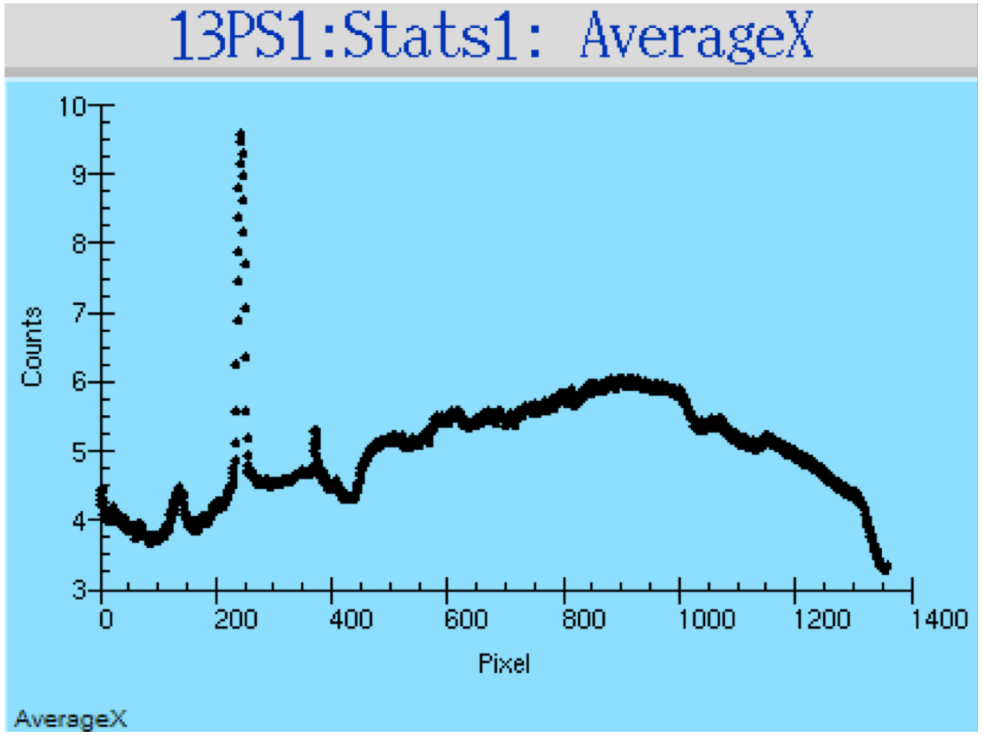


OVER1 offers 8 overlays:

1) Rectangle
2) Text "Hello"
3) …

NDPluginStats

- Computes min, max intensity etc.
- Computes profiles
- Advanced image statistics
  - Excess Kurtosis (flatness)
  - Skewness (symmetry)
  - Centroid & sigma

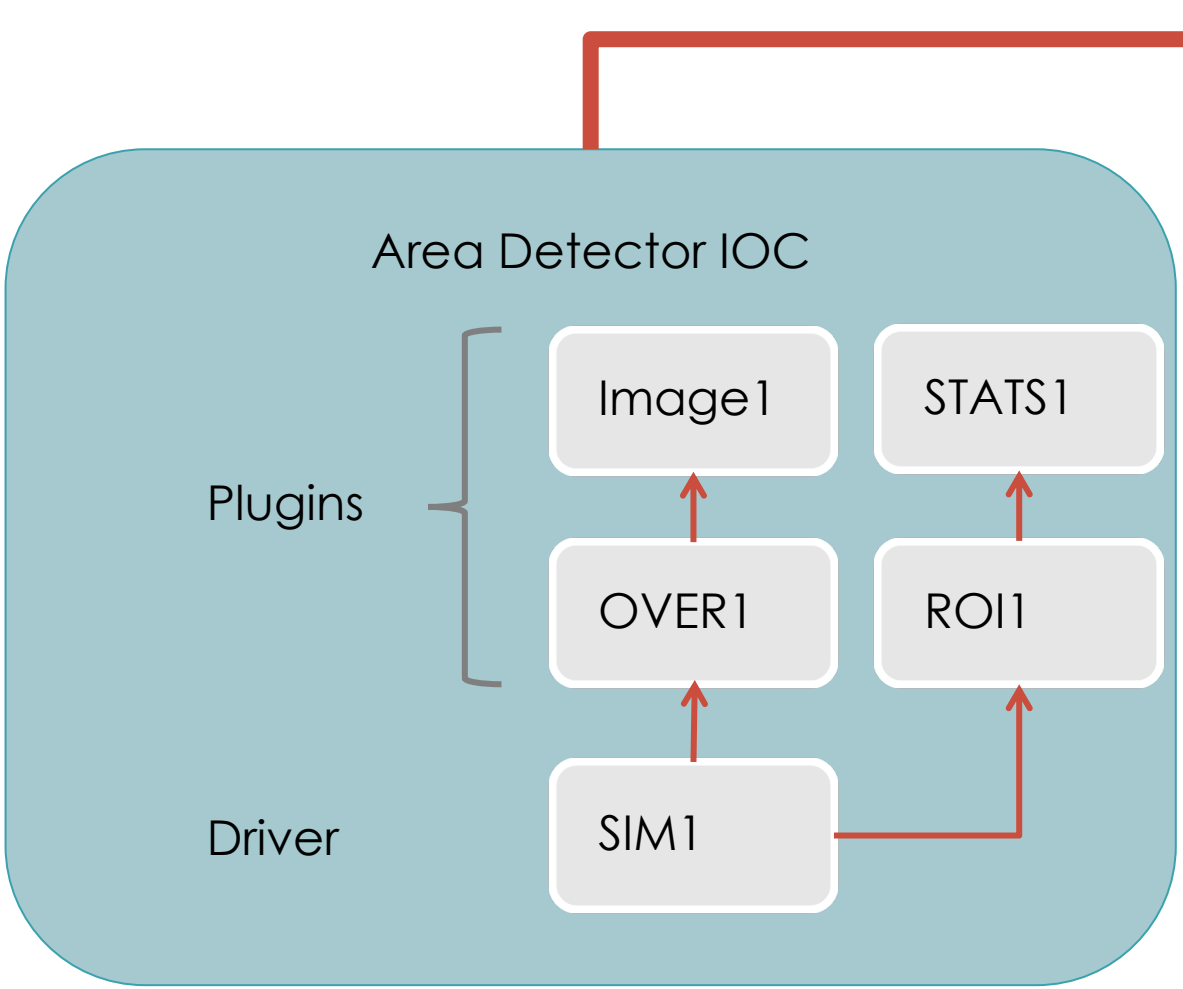


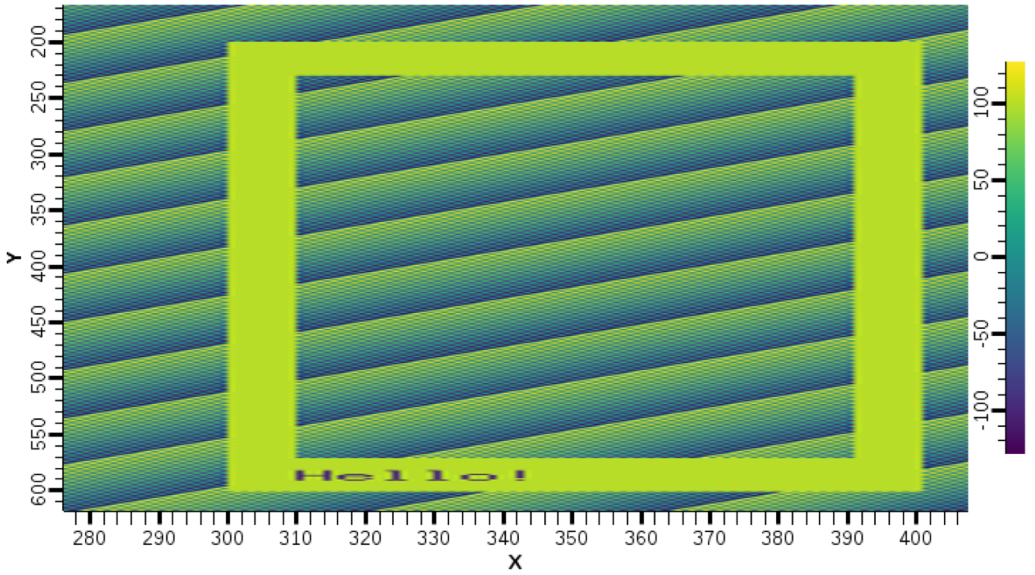
- On Detector, Plugins, All, find NDPluginStats “STATS1”, “More”
  - Set its Port to “SIM1”, Enable
  - Note how the Statistics show a min..max of 0..255

NDPluginROI

- Performs Region-Of-Interest calculations
  - Selects part of image
- On Detector, Plugins, All, find NDPluginROI “ROI1”, “More”
  - Set its Port to “SIM1”, Enable
  - Set X and Y size to 10, so ROI is small 10x10 corner of image
- On STAT1, change port from “SIM1” to “ROI1”
  - Note how the Statistics show a varying min..max as the image data rolls through that ROI

What we did


Area Detector IOC
Plugins
Image1
STATS1
OVER1
ROI1
Driver
SIM1


Hello!
Y
X
200
250
300
350
400
450
500
550
600
280
290
300
310
320
330
340
350
360
370
380
390
400
100
50
0
-50
-100

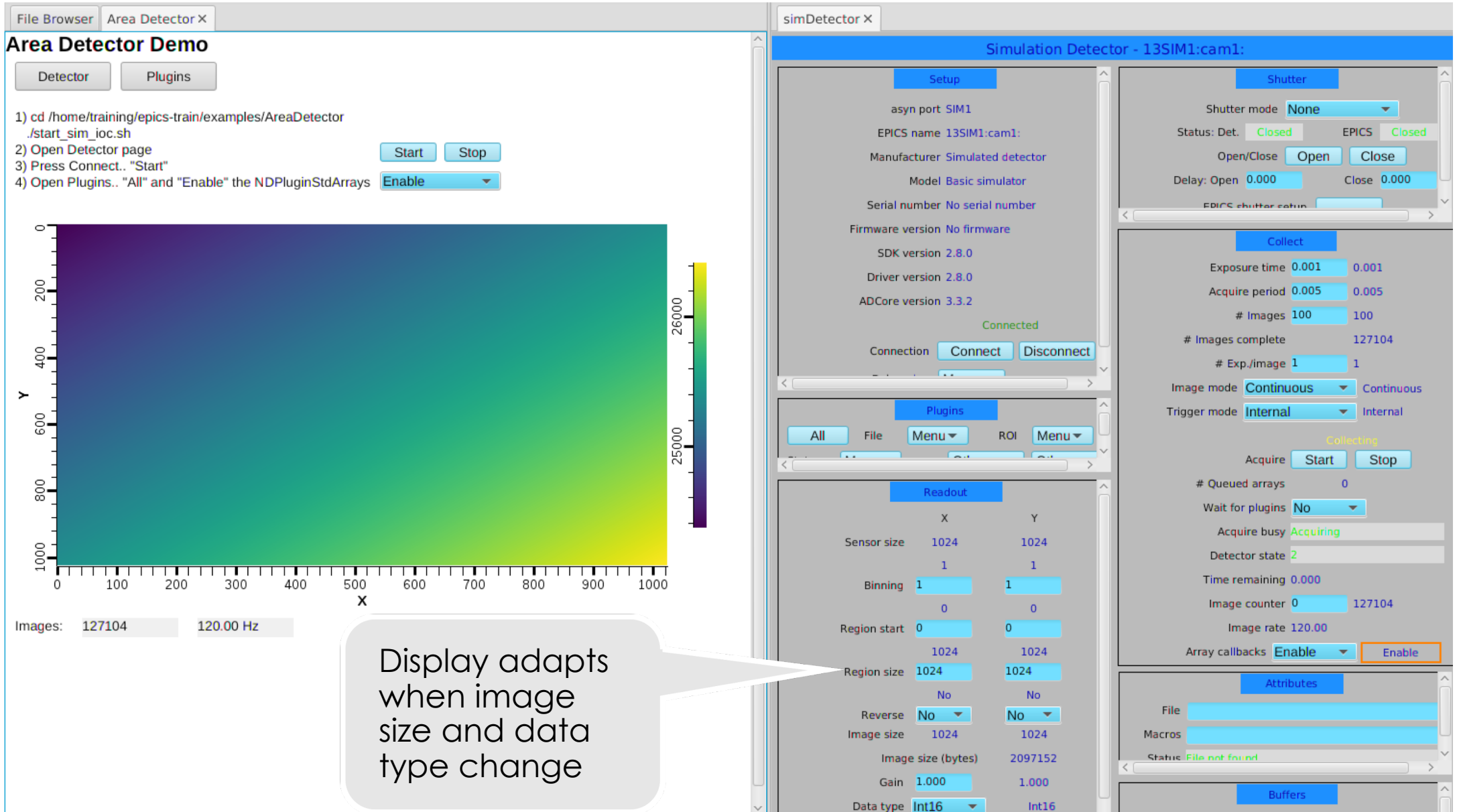
More Plugins

- Process
  - Background subtraction, clipping, recursive averaging over N images, ..
- Saving images in various formats
  - Adding data from PVs as “Attributes”
  - PNG, JPEG, TIFF, HDF5, …
- Serving NDArray via PVA
  - For displays: No need to configure size, data type, …
  - For ADPvAccess Driver: Process data on different hosts

NDPluginPVA – Serve PVA ‘Image’

- In Plugins, “PVA1”
  - Set its Port to “SIM1” or “OVER1”, Enable
- PVAccess Tests
  - pvlist
  - pvinfo 13SIM1:Pva1:Image
  - pvget -r 'field(dimension)' 13SIM1:Pva1:Image
- In Display
  - Use “Image” widget
  - Set PV
  - No need to configure data size, data type

NDPluginPVA – Serve PVA ‘Image’



Area Detector

Ecosystem for handling

- Cameras
- Detectors
- “Images” in EPICS