areaDetector: What’s New?

Mark Rivers
GeoSoilEnviroCARS, Advanced Photon Source
University of Chicago
Outline

• Last TWG areaDetector talk was 2009
  – Won’t try cover everything since then!
• Changes in the last year, R2-0, R2-1, R2-2
• New and improved drivers
• New and improved plugins
• Future plans
Moved areaDetector to github

• areaDetector was getting too big.
  – New releases being held up waiting for testing on one detector types, etc.

• Hard to collaborate with other sites using APS Subversion repository
  – git and github provide much better tools for multi-site collaborations

• Moved in December 2013, with R2-0 release in April 2014

• Split into 3 “core” repositories, and separate repositories for each detector
New Organization

- 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
- Can clone individual repositories, or clone the entire project with
git clone -recursive https://github.com/areaDetector.git
New Organization

Top-level areaDetector

- `andor3App/` builds only a driver library, and depends only on base and async.
- `iocs/` builds an application and depends on autosave, busy, etc. Can disable building this.
New or improved drivers

- ADAndor3 driver for sCMOS cameras from Andor
- ADLightField driver for Princeton Instruments cameras using their LightField application
- ADAndor added Shamrock spectrometer control for Andor cameras
- ADmarCCD added support for triggered acquisition using new high-speed (-HS) detectors from Rayonix
- ADPSL major rewrite of Photonic Sciences Limited driver to support new server features
- PICAM driver (ADPiCam)
  - Princeton Instruments cameras, including recent models
  - Controlled via PICAM C library
  - Written by John Hammonds
LightField driver
LightField driver
Detector drivers (continued)

• Generic GigE driver (aravisGigE)
  – Should work with any GigEVision compliant camera. From Tom Cobb at Diamond.
  – Controlled using the Aravis reverse-engineered GigEVision library

• QImaging driver (ADQImaging)
  – QImaging cameras.
  – Controlled using Qimaging SDK
  – Written by Arthur Glowacki

• ADPvAccess
  – Driver that receives NDArrays over EPICS V4
  – Allows plugins to run in an EPICS IOC on a different machine than the detector
  – Written by David Hickin from Diamond.
  – Another version by Bruno Martins from BNL will likely be part of ADCore R3-0.
Detector drivers (continued)

• Perkin Elmer Dexela driver (ADDexela)
  – For Perkin Elmer Dexela CMOS flat-panel detectors
  – Written by Mark Rivers

• Quantum Detectors Merlin driver (ADMerlin)
  – For Merlin Medipix3 detector
  – Written by Giles Knapp at Diamond

• ADFastCCD
  – Driver for LBNL FastCCD.
  – Written by Stuart Wilkins at NSLS-II
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
- Example: Model GS3-U3-23S6M
  - 1920 x 1200 global shutter CMOS
  - No smear • Distortion-free
  - Dynamic range of 73 dB
  - Peak QE of 76%
  - Read noise of 7e-
  - Max frame rate of 162 fps (~400 MB/S, 4X faster than GigE)
  - USB 3.0 interface
  - $1,295
  - Comparable to PCO Edge and Andor Zyla for 10X less money
New driver for Pixirad CdTe pixel array detector
Similar to Pilatus, but CdTe gives very high efficiency to 80 keV or more
2 energy thresholds, so 2 different energy images simultaneously
In detector pool
NDAttributes

- Detector drivers and plugins read files like this to add NDAttributes to the arrays
- Values can come from driver/plugin, EPICS PVs, or user-written functions
- Can be written to files, used to control downstream plugins, etc.

```xml
<Attributes>
  <!-- Attributes -->
  <Attribute name="AcquireTime" type="PARAM" source="ACQ_TIME" datatype="DOUBLE" description="Camera acquire time"/>
  <Attribute name="Comment1" type="PARAM" source="MODEL" datatype="STRING" description="Comment 1"/>
  <Attribute name="Comment2" type="PARAM" source="MODEL" datatype="STRING" description="Comment 2"/>
  <Attribute name="Comment3" type="PARAM" source="MODEL" datatype="STRING" description="Comment 3"/>
  <Attribute name="CameraDistance" type="EPICS_PV" source="13BMDG1:TIIFF1:FileTemplate" datatype="DOUBLE" description="Camera distance"/>
  <Attribute name="CameraModel" type="PARAM" source="MODEL" datatype="STRING" description="CameraModel"/>
  <Attribute name="CameraRotation" type="EPICS_PV" source="13BMD:m38.VELO" datatypes="DOUBLE" description="Camera rotation"/>
  <Attribute name="CameraY" type="EPICS_PV" source="13BMD:m34.VAL" datatypes="DOUBLE" description="Camera Y (focus) position"/>
  <Attribute name="CameraX" type="EPICS_PV" source="13BMD:m33.VAL" datatypes="DOUBLE" description="Camera X position"/>
  <Attribute name="CameraRotation" type="EPICS_PV" source="13BMD:m37.VAL" datatypes="DOUBLE" description="Camera rotation"/>
  <Attribute name="CameraDistance" type="EPICS_PV" source="13BMD:m37.VAL" datatypes="DOUBLE" description="Camera distance"/>
  <Attribute name="ExposureTime" type="EPICS_PV" source="13BMD:SIS1:LNEOutputWidth" datatypes="DOUBLE" description="SIS output width=exposure time"/>
  <Attribute name="ExposurePeriod" type="EPICS_PV" source="13BMD:SIS1:Dwell" datatypes="DOUBLE" description="SIS Dwell=exposure period"/>
  <Attribute name="LiftAX" type="EPICS_PV" source="13BMD:XAS:t1.EAX" datatypes="DOUBLE" description="Lift table AY"/>
  <Attribute name="LiftAY" type="EPICS_PV" source="13BMD:XAS:t1.EAY" datatypes="DOUBLE" description="Lift table AY"/>
  <Attribute name="LiftAZ" type="EPICS_PV" source="13BMD:XAS:t1.EAZ" datatypes="DOUBLE" description="Lift table Z"/>
  <Attribute name="LiftAZ" type="EPICS_PV" source="13BMD:XAS:t1.EAZ" datatypes="DOUBLE" description="Lift table Z"/>
  <Attribute name="LiftZ" type="EPICS_PV" source="13BMD:XAS:t1.EZ" datatypes="DOUBLE" description="Lift table Z"/>
  <Attribute name="LiftY" type="EPICS_PV" source="13BMD:XAS:t1.EY" datatypes="DOUBLE" description="Lift table Y"/>
  <Attribute name="LiftY" type="EPICS_PV" source="13BMD:XAS:t1.EY" datatypes="DOUBLE" description="Lift table Y"/>
  <Attribute name="LiftX" type="EPICS_PV" source="13BMD:XAS:t1.EX" datatypes="DOUBLE" description="Lift table X"/>
  <Attribute name="LiftX" type="EPICS_PV" source="13BMD:XAS:t1.EX" datatypes="DOUBLE" description="Lift table X"/>
  <Attribute name="RotationSpeed" type="EPICS_PV" source="13BMD:m89.VAL" datatypes="DOUBLE" description="Rotation speed"/>
  <Attribute name="SampleOmega" type="EPICS_PV" source="13BMD:m38.RBV" datatypes="DOUBLE" description="Sample omega"/>
  <Attribute name="SampleY" type="EPICS_PV" source="13BMD:m90.VAL" datatypes="DOUBLE" description="Sample Y centering"/>
  <Attribute name="SampleX" type="EPICS_PV" source="13BMD:m85.VAL" datatypes="DOUBLE" description="Sample X centering"/>
  <Attribute name="SampleYCent" type="EPICS_PV" source="13BMD:m89.VAL" datatypes="DOUBLE" description="Sample Y centering"/>
  <Attribute name="SampleXCent" type="EPICS_PV" source="13BMD:m91.VAL" datatypes="DOUBLE" description="Sample X centering"/>
  <Attribute name="SampleHeight" type="EPICS_PV" source="13BMD:m34.VAL" datatypes="DOUBLE" description="Sample Y centering"/>
  <Attribute name="SampleHeight" type="EPICS_PV" source="13BMD:m35.VAL" datatypes="DOUBLE" description="Sample Y centering"/>
  <Attribute name="SampleHeight" type="EPICS_PV" source="13BMD:m36.VAL" datatypes="DOUBLE" description="Sample Y centering"/>
  <Attribute name="SampleHeight" type="EPICS_PV" source="13BMD:m37.VAL" datatypes="DOUBLE" description="Sample Y centering"/>
  <Attribute name="SampleHeight" type="EPICS_PV" source="13BMD:m38.VAL" datatypes="DOUBLE" description="Sample Y centering"/>
  <Attribute name="SampleHeight" type="EPICS_PV" source="13BMD:m39.VAL" datatypes="DOUBLE" description="Sample Y centering"/>
  <Attribute name="SampleHeight" type="EPICS_PV" source="13BMD:m40.VAL" datatypes="DOUBLE" description="Sample Y centering"/>
  <Attribute name="SampleHeight" type="EPICS_PV" source="13BMD:m41.VAL" datatypes="DOUBLE" description="Sample Y centering"/>
  <Attribute name="SampleHeight" type="EPICS_PV" source="13BMD:m42.VAL" datatypes="DOUBLE" description="Sample Y centering"/>
  <Attribute name="SampleHeight" type="EPICS_PV" source="13BMD:m43.VAL" datatypes="DOUBLE" description="Sample Y centering"/>
  <Attribute name="SampleHeight" type="EPICS_PV" source="13BMD:m44.VAL" datatypes="DOUBLE" description="Sample Y centering"/>
  <Attribute name="SampleHeight" type="EPICS_PV" source="13BMD:m45.VAL" datatypes="DOUBLE" description="Sample Y centering"/>
  <Attribute name="SampleHeight" type="EPICS_PV" source="13BMD:m46.VAL" datatypes="DOUBLE" description="Sample Y centering"/>
  <Attribute name="SampleHeight" type="EPICS_PV" source="13BMD:m47.VAL" datatypes="DOUBLE" description="Sample Y centering"/>
  <Attribute name="SampleHeight" type="EPICS_PV" source="13BMD:m48.VAL" datatypes="DOUBLE" description="Sample Y centering"/>
  <Attribute name="SampleHeight" type="EPICS_PV" source="13BMD:m49.VAL" datatypes="DOUBLE" description="Sample Y centering"/>
  <Attribute name="SampleHeight" type="EPICS_PV" source="13BMD:m50.VAL" datatypes="DOUBLE" description="Sample Y centering"/>
  <Attribute name="SampleHeight" type="EPICS_PV" source="13BMD:m51.VAL" datatypes="DOUBLE" description="Sample Y centering"/>
  <Attribute name="SampleHeight" type="EPICS_PV" source="13BMD:m52.VAL" datatypes="DOUBLE" description="Sample Y centering"/>
  <Attribute name="SampleHeight" type="EPICS_PV" source="13BMD:m53.VAL" datatypes="DOUBLE" description="Sample Y centering"/>
  <Attribute name="SampleHeight" type="EPICS_PV" source="13BMD:m54.VAL" datatypes="DOUBLE" description="Sample Y centering"/>
  <Attribute name="SampleHeight" type="EPICS_PV" source="13BMD:m55.VAL" datatypes="DOUBLE" description="Sample Y centering"/>
  <Attribute name="SampleHeight" type="EPICS_PV" source="13BMD:m56.VAL" datatypes="DOUBLE" description="Sample Y centering"/>
  <Attribute name="SampleHeight" type="EPICS_PV" source="13BMD:m57.VAL" datatypes="DOUBLE" description="Sample Y centering"/>
  <Attribute name="SampleHeight" type="EPICS_PV" source="13BMD:m58.VAL" datatypes="DOUBLE" description="Sample Y centering"/>
  <Attribute name="SampleHeight" type="EPICS_PV" source="13BMD:m59.VAL" datatypes="DOUBLE" description="Sample Y centering"/>
  <Attribute name="SampleHeight" type="EPICS_PV" source="13BMD:m60.VAL" datatypes="DOUBLE" description="Sample Y centering"/>
NDPluginCircularBuff

- Buffers NDArrays in a circular buffer.
- 2 NDAttributes can be used in trigger equation.
- Outputs the arrays when trigger calculation is true, or when forced.
- Supports pre-trigger and post-trigger samples
- Written by Alan Greer at Observatory Sciences
NDPluginAttribute

- Extracts NDAttributes from an NDArray and publishes as scalar and time-series arrays
- Written by Matt Pearson at ORNL
NDPluginROIStat

- Supports multiple regions-of-interest with simple statistics on each.
- More efficient and convenient than using NDPluginROI and NDPluginStats when many regions of interest with simple statistics are needed.
- Written by Matthew Pearson at ORNL
NDPluginTransform improvements

- Greatly simplified: just 8 operations including null operation
- 13-85 times faster than previous releases depending on data type, color mode
- Added support for text overlays (Keith Brister, LS-CAT)
- Added support for line widths in cross and rectangle overlays (Matt Pearson, ORNL)
Other Recent Plugins

• ffmpegServer
  – MJPEG server that allows viewing images in a Web browser. From DLS.
  – Puts compressed images on the network, greatly reducing bandwidth compared to uncompressed channel access arrays.
  – Written by Tom Cobb from Diamond

• ADPvAccess
  – Plugin that sends NDArrays over EPICS V4
  – Allows plugins to run in an EPICS IOC on a different machine than the detector
  – Written by David Hickin from Diamond.

• ADPluginEdge
  – Does edge detection using the OpenCV Canny function
  – Written by Keith Brister at LS-CAT
NDPluginFile

New
NDPluginFile Recent Features

• File plugins can now create directories
  – `CreateDirectory` record controls whether directories are created if they don't exist.
    • Zero (default), no directories are created.
    • Negative, then absolute value is the maximum of directories that will be created (i.e. -1 will create a maximum of one directory to complete the path, -2 will create a maximum of 2 directories).
    • Positive, then at least that many directories in the path must exist (i.e. a value of 1 will create all directories below the root directory and 2 will not create a directory in the root directory).
NDPluginFile Recent Features

• “Lazy-open”
  – Normally files in stream mode are opened when Capture PV is set to 1
  – This requires that there have already been an NDArray received by that plugin with the correct dimensions and attributes
  – “Lazy-open” is selected the file is not opened until the first NDArray callback happens after Capture is set to 1.
  – Simpler for users, but poorer performance, can lead to dropped arrays

• File plugins can write files with a temporary suffix and then rename the file after writing is complete.
  – Allows rsync, etc. to be used to copy files, with guarantee that they are complete
File Plugin Enhancements

• NDFileTIFF
  – Supports any NDArray data type
  – Stores NDAttributes as ASCII user tags, up to 490.

• NDFileHDF5
  – Now supports using an XML file to define the layout and placement of NDArrays and NDAttributes in the HDF5 file.
  – Can be used to create NeXus-compliant files without using NDFileNeXus plugin
  – NDFileNeXus will probably be deprecated in a future release.
  – Major project, collaboration of APS and Diamond (Arthur Glowacki, Ulrik Pedersen, Alan Greer).
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>
</group>
<group name="NDAttributes" ndattr_default="true">
<attribute name="NX_class" source="constant" value="NXcollection" type="string"></attribute>
</group>
<group name="performance">
<dataset name="timestamp" source="ndattribute"></dataset>
</group>
</group>
</group>
<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>
</group>
</group>
</xml>
Other Recent Enhancements

- Added new attribute type, NDAttrSourceFunct. This type of attribute gets its value from a user-defined C++ function. It can thus be used to get any type of metadata. Previously only EPICS PVs and driver/plugin parameters were available as metadata.

- $(P)$(R)ADCoreVersion_RBV provides version of ADCore. This allows CA clients to alter their behavior depending on version of ADCore used to build the plugin or driver.

- simDetectorNoIOC
  - Example standalone C++ application that instantiates a simDetector without running an EPICS IOC
  - Shows that areaDetector drivers and plugins only depend on libCom and asyn libraries. Can be used from other control systems.
Future Ideas (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
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
Future Ideas

• Extend areaDetector concepts to other types of detectors:
  – ADCs
  – Electrometers
  – Waveform digitizers
  – Oscilloscopes?

• They all produce 1-D (or 2-D for multi-channel inputs) arrays that could benefit from plugins for file saving, FFTs, ROI extraction, digital filtering, etc.

• We can’t currently use the areaDetector file plugins to save MCA data, for example.
Future Ideas

• Export NDArrays via EPICS V4

• David Hickin (DLS) has demonstrated:
  – A plugin that exports NDArrays as V4 objects over Channel Access
  – An ADDriver that receives the V4 objects on another machine and has its own set of plugins

• Allows using multiple machines, and multiple processes, not just multiple cores in a single IOC for plugin processing
areaDetector Collaboration

• The move to GitHub has really helped areaDetector become a collaborative effort
• Many more people are contributing via additions and bug fixes.
• Make changes in their fork on github and then issue a “pull request”.
• Collaboration meeting ~monthly on Google Hangout (U. Pedersen, M. Rivers, A. Glowacki, M. Pearson, M. Kraimer, N. Rees, D. Hickin, T. Cobb)
• In-person meetings ~2 times/year.
• Developed a road-map, following it pretty well.