Andy Salnikov
2011-06-21
SLAC Scientific Computing Workshop

Analysis Frameworks for LCLS

Motivation

- LCLS produces data at scale of petabyte(s) per year

- Our users have very diverse background and use different analysis tools:

  - Matlab, IDL, C/C++, Python, HDF5

- Traditional tools not adequate for LCLS data volumes

- New analysis tools are necessary

  - Developed and supported by SLAC

  - Performance and efficient resource utilization is critical

  - Simple but flexible

First-generation analysis tools

- Off-line analysis effort started after first data appeared

- First versions of the analysis frameworks were an interim solution:

  - C++ on-line analysis tool (myana) by on-line group

  - Python-based off-line analysis tool (pyana)

Pyana framework

- Python wrapper for C++ data classes

- Input from XTC (raw) data files

- Python numeric packages: numpy, scipy, matplotlib

- User extension modules written in Python

- Performance OK if no Python loops

- Simple to learn, but not all groups have time to invest in Python
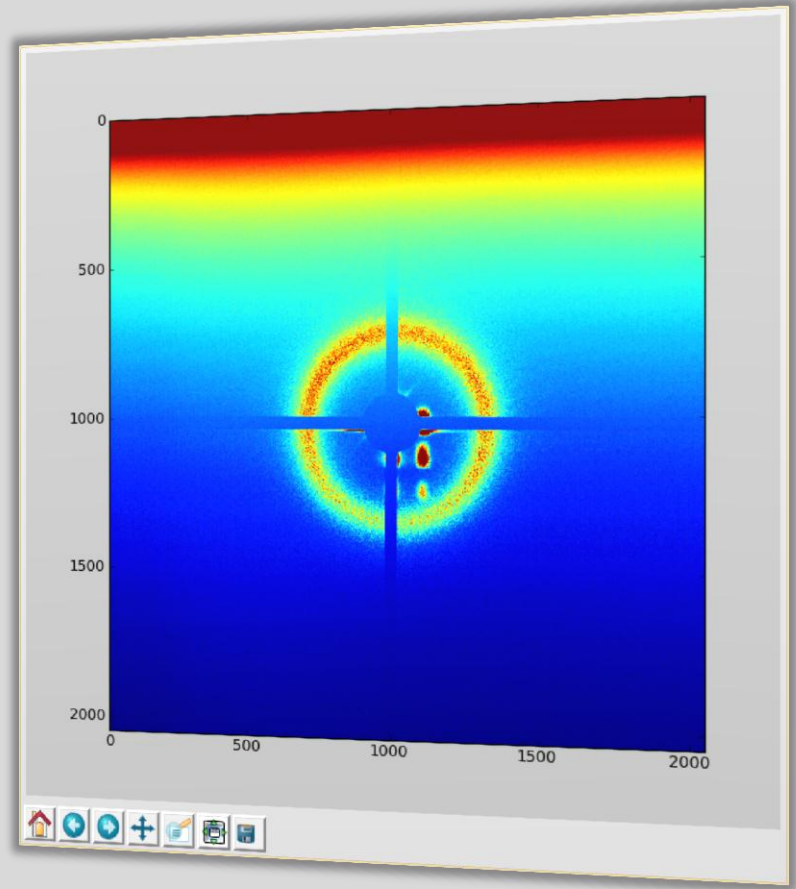
New generation of frameworks

- Development of new C++-based framework started 1Q 2011

- Will accommodate various data formats: XTC and HDF5, transparent to user code

- User modules written in C++, job control options defined in simple configuration file

- Library of chainable re-useable modules

- Histogramming package based on ROOT

Advanced framework features

- Planned multi-core support

  - Event-level parallelism

  - Multi-threaded libraries of analysis algorithms

- Python wrapper for Framework and data classes

  - Users can choose writing analysis modules in C++ or Python

- Interactive features to help integration with GUI and event displays

Status

- First version of new framework is released

    - Input from XTC (raw) files only

    - Small set of example modules

    - Work in progress on HDF5 input, set of standard modules for calibration and filtering