Scaling smalldata_tools for LCLS-II

Vincent Esposito and Silke Nelson, LCLS Experiment Control System

Oct 21, 2025

SLAC NATIONAL ACCELERATOR LABORATORY

Stanford University

U.S. DEPARTMENT of ENERGY

What is smalldata_tools?

From data acquisition to user files

- Configurable pre-processing tool built on top of psana. Two approaches:
  - Event-based reduction
  - Binned reduction
- Output files (hdf5) suited for Jupyter Notebooks analysis / exploration
- Abstract event loop, calibrations, scaling from users
- Used at most instruments at LCLS
- **Users can focus on their science rather than the computing and data processing**

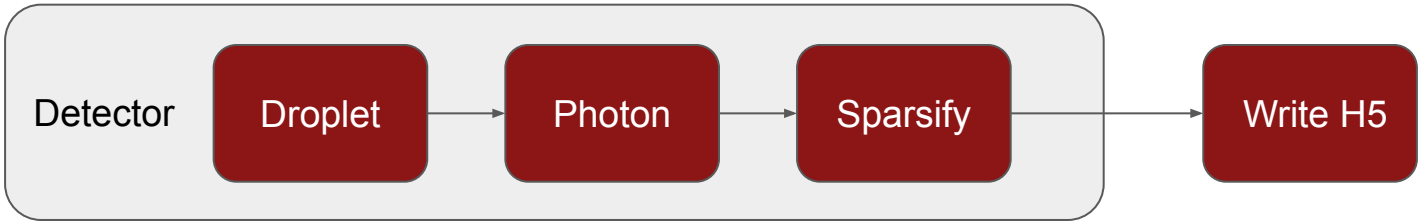
Approach 1: event-based reduction

Keep all events, but make them small

- Area and waveform detectors:
  - Region of interest, projection
  - Azimuthal integration
  - Droplet / photonization
  - Autocorrelation
  - … (easily extendable)
- Default set of scalar detector

Simple pipeline example: droplet + greedy-guess photon finding algorithms.

```
# Instantiate detector
det = DetObject(detname, run, **det_kwargs)

# Instantiate the different analysis functions
Roi_func = ROIFunc(**roi_args)
droplet_func = dropletFunc(**droplet_args)
photon_func = droplet2Photons(**photon_args)
sparsify = sparsifyFunc()

# Build a simple pipeline and add it to the detector
photon_func.addFunc(sparsify)
droplet_func.addFunc(photon_func)
roi_func.addFunc(droplet_func)
det.addFunc(roi_func)
```



Approach 2: cube

Aggregate events, but keep full detectors

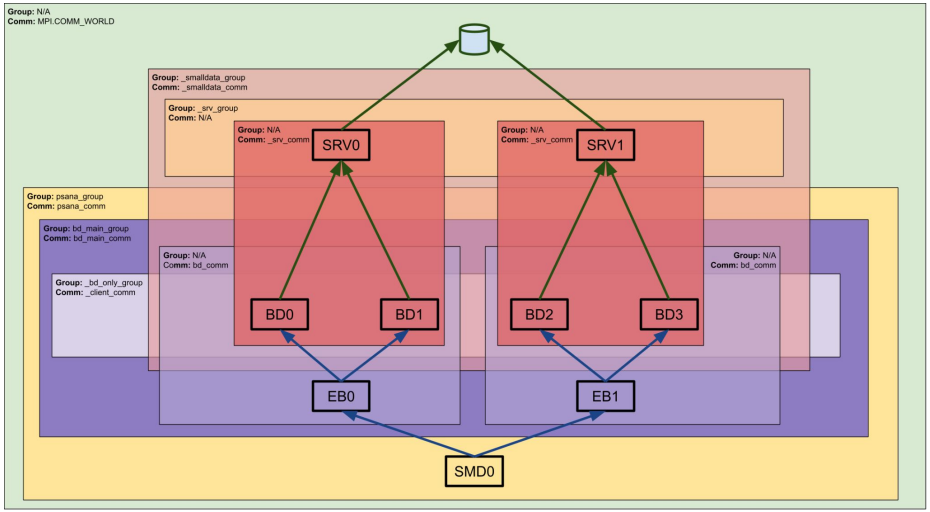
- Event aggregation:
  - Step scans
  - User-defined multi-dimensional binning
- Event screening and filtering
  - Laser on / off
  - Discard / filter events

```python
def screener(run):
    ddets = hutch_default.rixDetectors(run)
    lightstatus_det = [d for d in ddets if d.name == "lightStatus"][0]

    # Individual filters
    laser_on = BoolFilter(
        lightstatus_det, "laser", expected_state=True, label="laser_on"
    )

    laser_off = BoolFilter(
        lightstatus_det, "laser", expected_state=False, label="laser_off"
    )

    xray_on = BoolFilter(
        lightstatus_det,
        "xray",
        expected_state=True,
    )

    xray_off = BoolFilter(
        lightstatus_det, "xray", expected_state=False, label="dropped_shots"
    )

    # Combine individual filters into composite filters for different states
    screener_on = CompositeFilter([laser_on, xray_on])
    screener_off = CompositeFilter([laser_off, xray_on])

    # Combine all screeners into an OR screener to be run by the cube
    event_screener = CompositeFilter(
        [screener_on, screener_off, xray_off], require_all=False
    )
    return event_screener
```

Scaling: psana

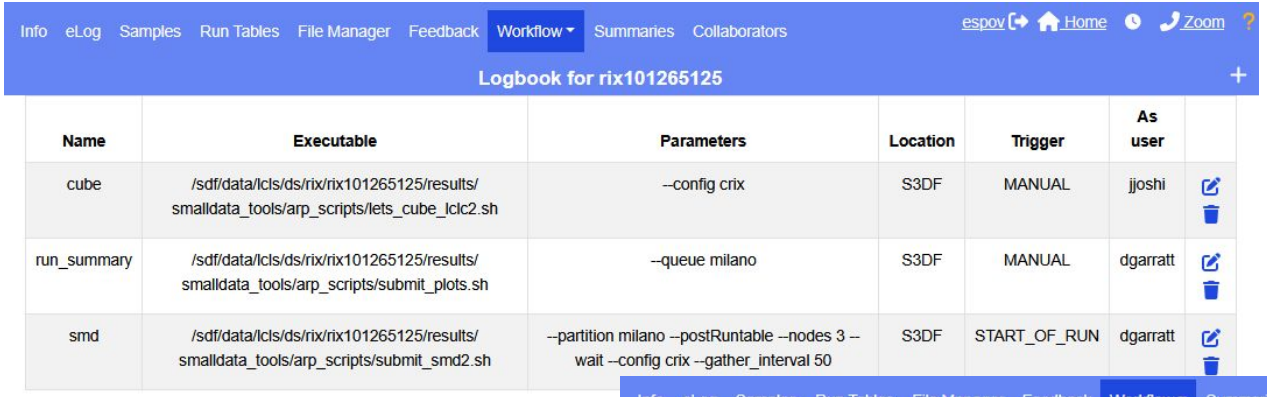
MPI communicators and groups in psana

- LCLS-1 psana: embarrassingly parallel
- LCLS-2 psana: complex parallelization with multiple stages:
  - SMD0: distribute block of events
  - EB (event builder)
  - BD (big data)
  - SRV: writers
- Knobs to tune each stage of the parallel processing



Run with slurm: a simple interface

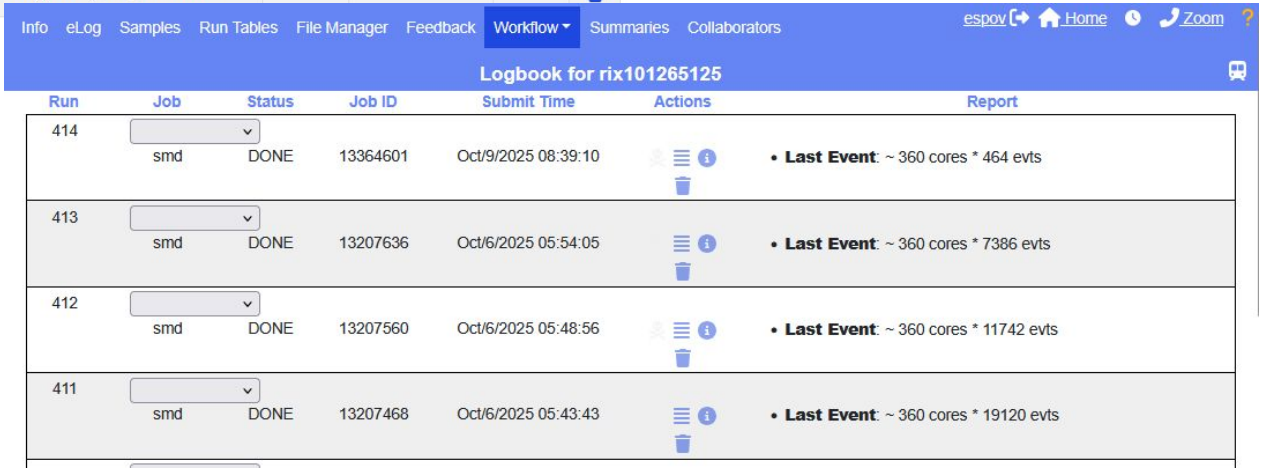
The Automated Run Processing (ARP)



Name	Executable	Parameters	Location	Trigger	As user
cube	/sdf/data/lcls/ds/rix/rix101265125/results/smalldata_tools/arp_scripts/lets_cube_lclc2.sh	--config crix	S3DF	MANUAL	jjoshi
run_summary	/sdf/data/lcls/ds/rix/rix101265125/results/smalldata_tools/arp_scripts/submit_plots.sh	--queue milano	S3DF	MANUAL	dgarratt
smd	/sdf/data/lcls/ds/rix/rix101265125/results/smalldata_tools/arp_scripts/submit_smd2.sh	--partition milano --postRuntable --nodes 3 --wait --config crix --gather_interval 50	S3DF	START_OF_RUN	dgarratt

Submit slurm jobs from the elog web interface. No CLI needed.

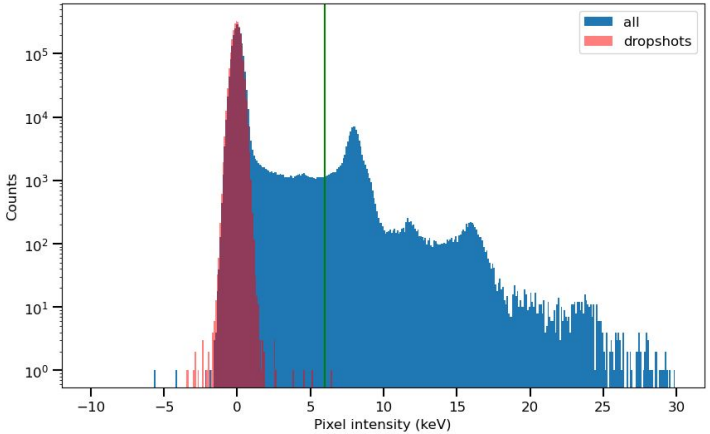
Processed files available minutes after data run is over

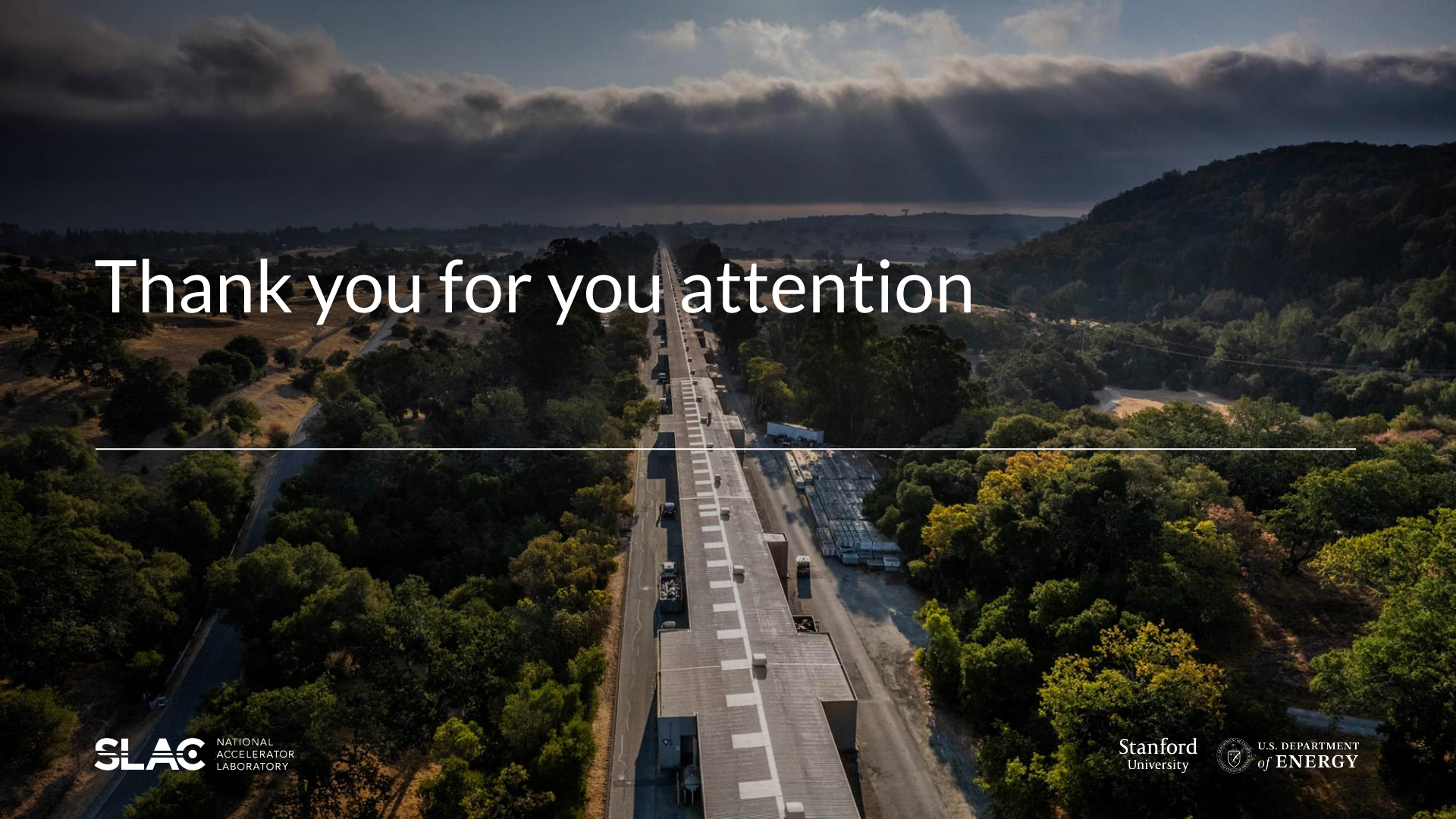


Run	Job	Status	Job ID	Submit Time	Actions	Report
414	smd	DONE	13364601	Oct/9/2025 08:39:10		• **Last Event**: ~ 360 cores * 464 evts
413	smd	DONE	13207636	Oct/6/2025 05:54:05		• **Last Event**: ~ 360 cores * 7386 evts
412	smd	DONE	13207560	Oct/6/2025 05:48:56		• **Last Event**: ~ 360 cores * 11742 evts
411	smd	DONE	13207468	Oct/6/2025 05:43:43		• **Last Event**: ~ 360 cores * 19120 evts

Challenges and outlook

- LCLS welcomes users with a range of computing skills
- Build trust: more complex systems become black-boxes to the end users
- Balance between stability / continuity and versatility




Thank you for you attention
SLAC NATIONAL ACCELERATOR LABORATORY
Stanford University
U.S. DEPARTMENT of ENERGY