

Pipeline Runtime Analysis

- ▶ Total runtime: 26.91s
- ▶ Main stages:
 - ▶ **LoadData: 11.40s (42.3%)**
 - ▶ Energy threshold: 11.38s
 - ▶ Delay binning: 0.02s
 - ▶ **MakeHistogram: 11.27s (41.9%)**
 - ▶ Histogram calculation: 11.27s with JIT compiler (down from > 60s unoptimized)
 - ▶ **Signal Analysis: 2.72s (10.1%)**
 - ▶ EMD + P-values + Masks: 1.20s
 - ▶ PumpProbe: 1.52s

Key Bottlenecks

- ▶ Energy threshold computation in LoadData
- ▶ Histogram calculation with large frame count (37,301)

Data loading differences

- ▶ Two distinct event selection and delay calculation workflows for different datasets
- ▶ Key differences in:
 - ▶ HDF5 path structures
 - ▶ Event classification methods
 - ▶ Detector mask handling
 - ▶ Filter parameters

HDF5 Path Structure - Delay Encoding

Dataset A: xppx1003221

- ▶ Uses enc/lasDelay combined with timetool position
- ▶ Additional timing corrections applied

Dataset B: xppl1030522

- ▶ Uses enc/lasDelay2 directly
- ▶ Different delay calculation formula

Impact: Different delay calculations affect time binning and resolution

Event Classification

Dataset A:

```
# HDF5 attributes:
/evr/code_90          # uint8 array
/evr/code_91          # uint8 array
/evr/code_40          # uint8 array

# Processing:
laser_on = evr.code_90 == 1
laser_off = evr.code_91 == 1
```

Event Classification (cont.)

Dataset B:

```
# HDF5 attributes:
/lightStatus/laser      # bool array
/lightStatus/xray       # bool array
/lightStatus/valid      # bool array

# Processing:
laser_on = lightStatus/laser == True
xray_on = lightStatus/xray == True
```

Detector Mask Handling

Dataset A:

```
# HDF5 attributes:  
/UserDataCfg/jungfrau1M/ROI_0__ROI_0_ROI  
/UserDataCfg/jungfrau1M/mask  
  
# Processing:  
idx_tile = ROI_0__ROI_0_ROI[0,0]  
mask = jungfrau1M.mask[idx_tile][roi_slice_y,  
    roi_slice_x]
```

Detector Mask Handling (cont.)

Dataset B:

```
# HDF5 attributes:  
/UserDataCfg/jungfrau1M/ROI_0__ROI_0_mask  
  
# Processing:  
roi0_mask = ROI_0__ROI_0_mask[0]
```

Impact: Different approaches to background subtraction and signal isolation

Filter Parameters

IPM Position Filters:

- ▶ **Dataset A (dynamically calculated):**
 - ▶ X: $[-0.25, 0.45]$
 - ▶ Y: $[-0.6, 0.8]$
- ▶ **Dataset B:(hardcoded in script parameters)**
 - ▶ X: $[-0.45, 0.45]$
 - ▶ Y: $[-1.6, 0.0]$

TimeTool Integration

Dataset A

- ▶ Always enabled
- ▶ Fixed threshold

```
# Fixed threshold in
filters
filters['tt_amp'] = [0.015,
inf]
```

```
# Skip tt filter for
laser-off
if 'tt' not in key:
    laser_off_mask = mask
and filt
```

Dataset B

- ▶ Optional usage
- ▶ Configurable threshold

```
# Optional timetool usage
if use_timetool:
    filters['tt_amp'] =
[0.0, inf]
```

```
# Laser-on events only
if key == 'tt_amp':
    laser_on_mask = mask
and filt
```