# Dranspose

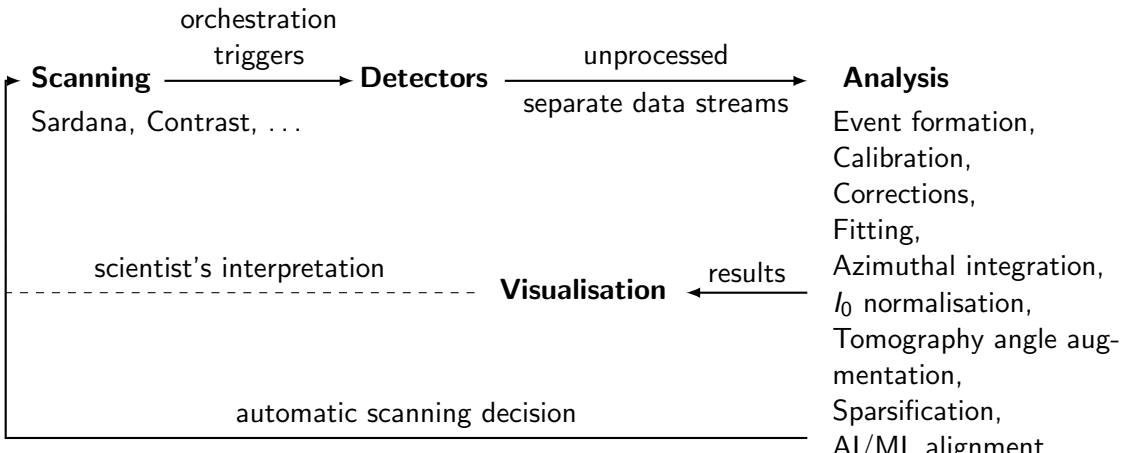
## A constrained map-reduce live analysis pipeline for fast experimental feedback

Felix Engelmann, Scientific Data

fe-research@nlogn.org

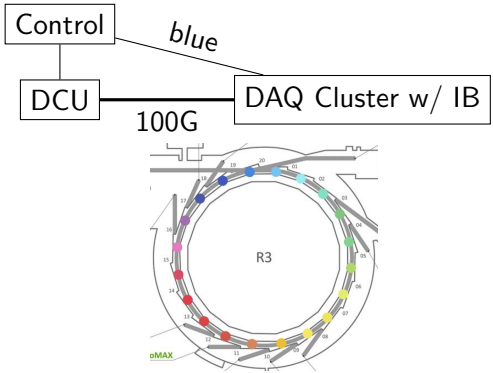
7th October 2024

# Experimental Feedback Loop



# Storage Architecture at MAX IV

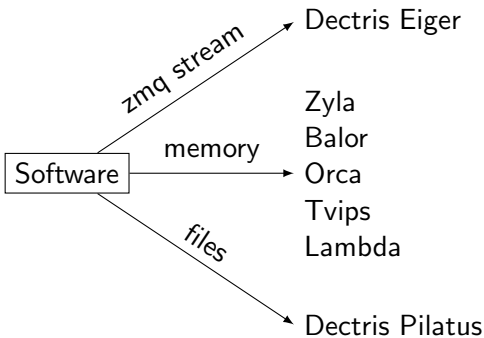
- Infiniband is expensive
- No GPFS at beamlines
- NFS mounts for viewing




```
GPFS cluster information
========================
  GPFS cluster name:         daq.maxiv.lu.se
  GPFS cluster id:           1615490..
  GPFS UID domain:           daq.maxiv.lu.se
  Remote shell command:      /usr/bin/ssh
  Remote file copy command:  /usr/bin/scp
  Repository type:           CCR

 Node  Daemon node name    IP address      Admin node name     Designatio
-------------------------------------------------------------------------
   1   p-daq-cn-4          172.18.10.104   p-daq-cn-4          quorum-mar
   2   p-daq-cn-3          172.18.10.103   p-daq-cn-3          quorum-mar
   8   p-daq-cn-2          172.18.10.102   p-daq-cn-2          quorum-mar
  17   p-jupyter-0         172.18.1.240    p-jupyter-0         perfmon
  18   p-jupyter-1         172.18.1.241    p-jupyter-1         perfmon
  19   p-jupyter-a100-0    172.18.1.250    p-jupyter-a100-0    perfmon
  20   p-jupyter-a100-1    172.18.1.251    p-jupyter-a100-1    perfmon
  21   p-compr-0           172.18.10.80    p-compr-0           perfmon
  22   p-daq-k8s-kirk-2    172.18.10.227   p-daq-k8s-kirk-2    perfmon
  23   p-daq-k8s-kirk-3    172.18.10.228   p-daq-k8s-kirk-3    perfmon
  24   p-daq-k8s-kirk-4    172.18.10.229   p-daq-k8s-kirk-4    perfmon
  25   p-daq-k8s-picard-2  172.18.10.224   p-daq-k8s-picard-2  perfmon
  26   p-daq-k8s-picard-3  172.18.10.225   p-daq-k8s-picard-3  perfmon
  27   p-daq-k8s-picard-4  172.18.10.226   p-daq-k8s-picard-4  perfmon
  29   p-daq-k8s-infra-p0  172.18.10.240   p-daq-k8s-infra-p0  perfmon
  30   p-daq-k8s-infra-k0  172.18.10.241   p-daq-k8s-infra-k0  perfmon
  31   p-dqd-picard-0      172.18.10.17    p-dqd-picard-0      perfmon
  32   p-dqd-kirk-0        172.18.10.16    p-dqd-kirk-0        perfmon
  33   p-daq-k8s-formax-0  172.18.10.230   p-daq-k8s-formax-0  perfmon
  34   p-daq-k8s-kirk-0    172.18.10.242   p-daq-k8s-kirk-0    perfmon
  35   p-daq-k8s-picard-0  172.18.10.244   p-daq-k8s-picard-0  perfmon
  36   p-daq-k8s-kirk-1    172.18.10.243   p-daq-k8s-kirk-1    perfmon
```

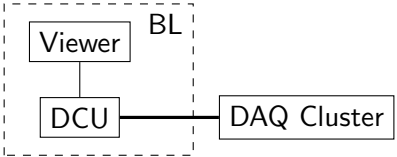
# Unified Streaming



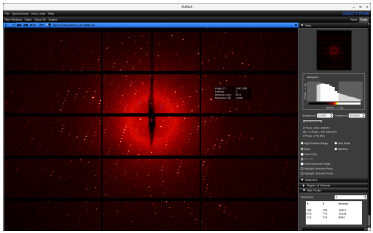

## Streamer

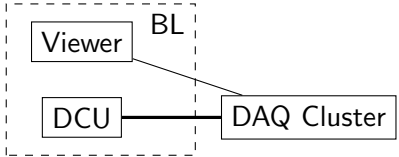
- Take frame buffers from SDK
- Opt. read files from local disk
- Wrap data in zmq frames
- Send to receiver

# Live Viewing

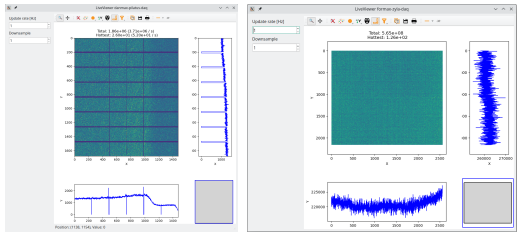



- Often provided by vendor
- Just works™
- Local network load

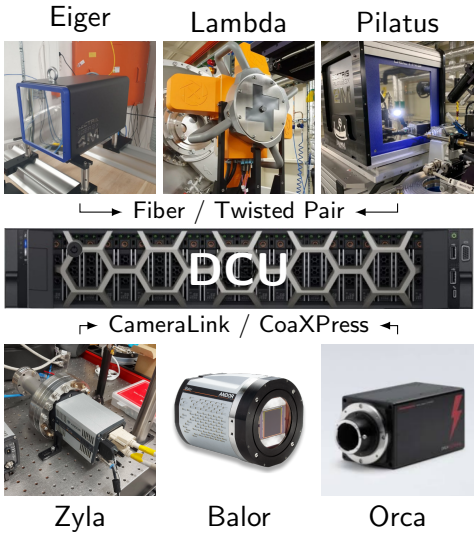




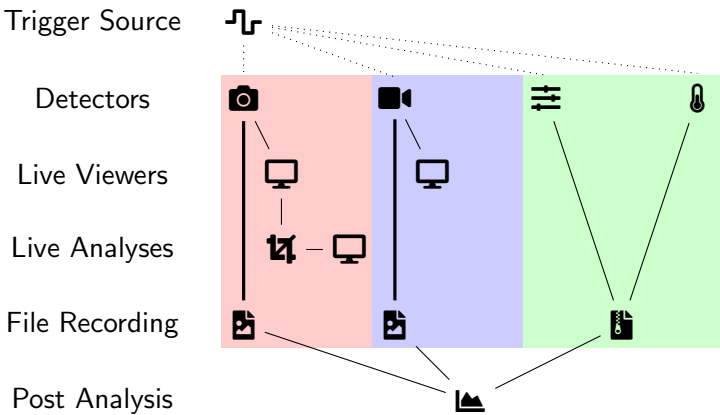

- More network load
- Reusable and common to all 2d
- Fast integration of new detectors
- Heavy processing possible



# High-Bandwidth DAQ scheme



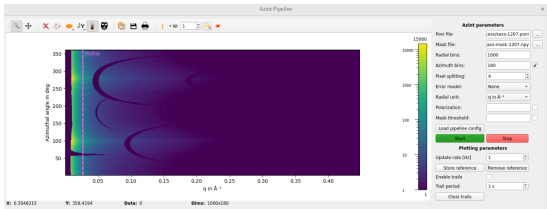
# Most Existing Infrastructure

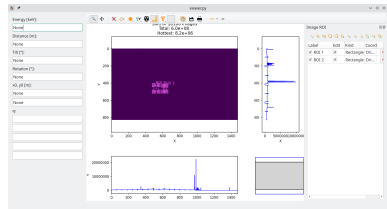


# Limitations of Live Analysis

## Single Data Stream

- only one detector data available
- simple tools, e.g. azint, crop, time integration




## Beyond Limits

normalisation to $I_0$, sorting by motor position

# Additional Data Sources

## Devices

PandAbox

XSpress3

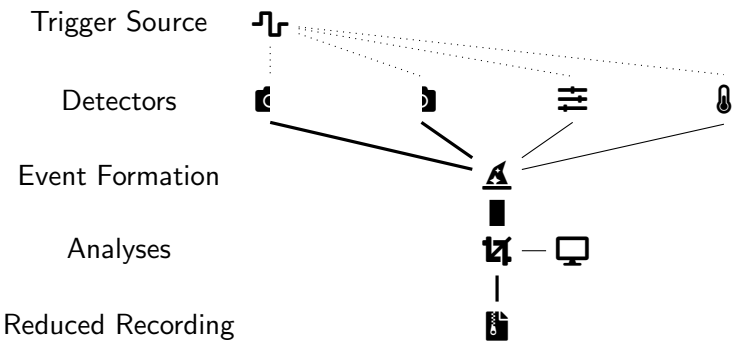
Basler


AlbaEM

## Software

- Sardana
  - Icepap
  - Meta-data
  - Filename
  - Snapshots
- Contrast
- Bluesky

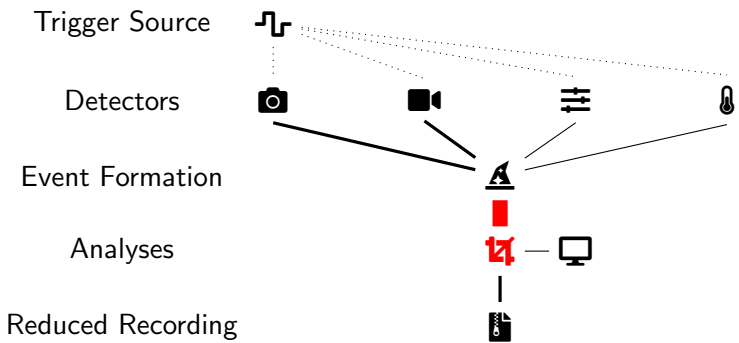
# Event Formation




| $t \downarrow$ | | | | |
|---|---|---|---|---|
| e1 | | | 0.5 | |
| e2 | | | | |
| e3 | | | 0.6 | |
| e4 | | | | |
| e5 | | | 0.5 | |
| e6 | | | | |

- non-uniform triggers

# Event Formation Bottleneck



# Existing Solutions

## Apache Flink

- exactly-once guarantee
- many small events

## Hummingbird

- XFEL development
- tailored to pulsed facilities

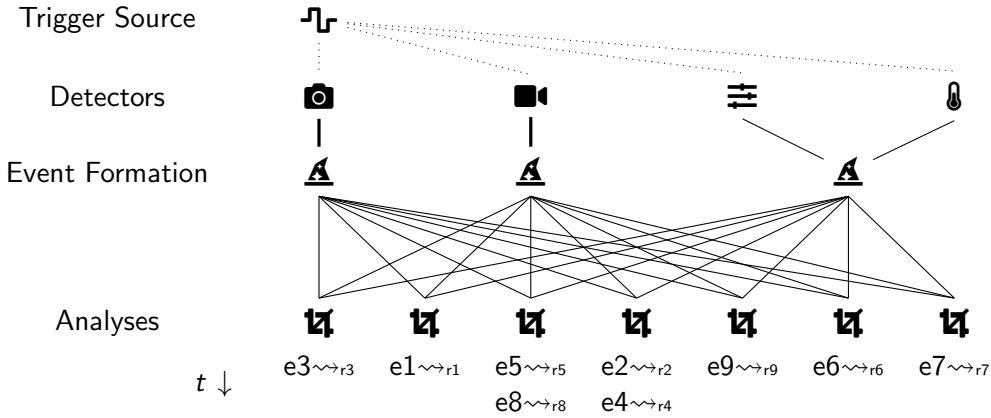
## OnDA / HiDRA

- DESY developments
- maintained?

## EWOKS

- large ecosytem
- data ingress from blissdata

# Parallelisation by Event, not Stream (`map`)




## Balancing Constraints

- event $2n$ and $2n+1$ to same worker
- event $m$ to worker `debug`

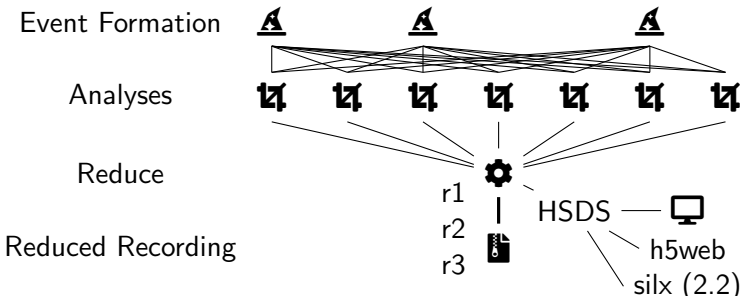
## Missing Inter Event Analysis

- time integration
- long temporal correlations

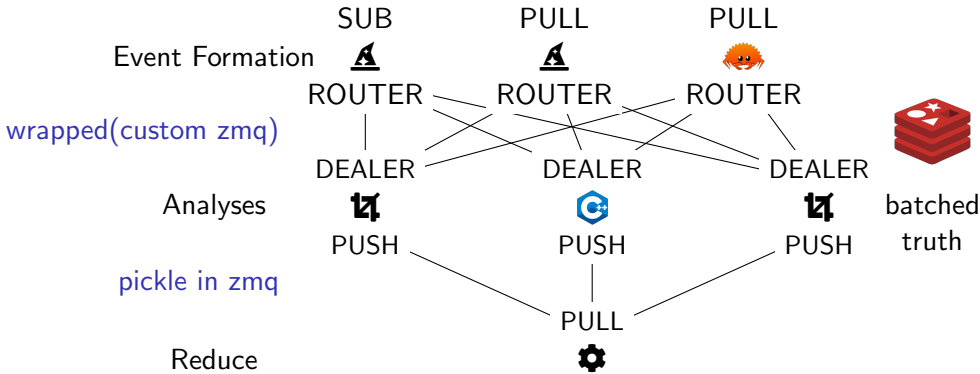
# Sequential Reduce

## Operations at Acquisition Speed

- append to list
- sum



# Behind the Scenes




pytest

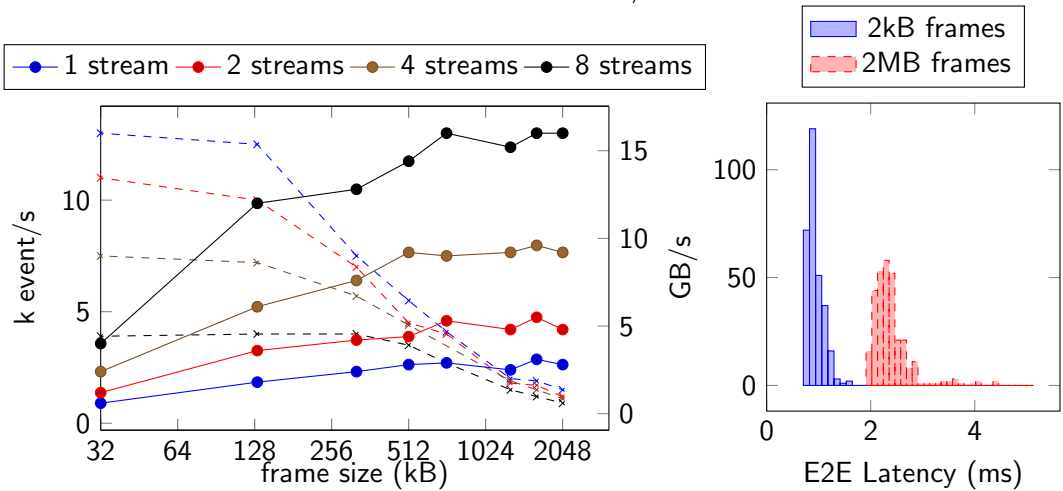
- end-to-end scans
- includes `rust` component

Typing

- MyPy
- Pydantic

# Performance Estimates

K8s cluster of 10: dual Xeon® Gold 6326, 100 Gbit/s Mellanox SR-IOV, 465 GB RAM




limits on around **2 GB/s** per stream or **12 kHz** at a **few ms**

# Development: Testing

## Recording

- ingesters optionally write all streams to disk (sequential cbor dumps)
- common experimental setup

## Replay

- from recorded stream dumps
- from hdf5 files

## Local

run file-based ingesters, workers and reducer locally

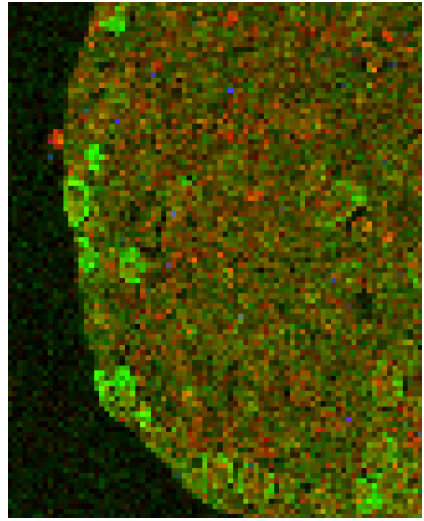
# Case Study: NanoMAX

## Sources

- XSpress3
- Contrast (Motor Positions)

## Analysis / Output

- PyMCA fluorescence fitting
- Concentration Map



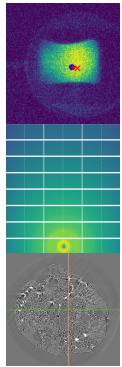
# Case Study: DanMAX

Sardana Basler Pilatus Orca PandAbox

Beam CoG, Beam Alignment

Pileup File

Encoder angle



```
python -m pip install dranspose
docker-compose, helm
```

https://dranspo.se