Data Challenges at the European X-ray Free Electron Laser

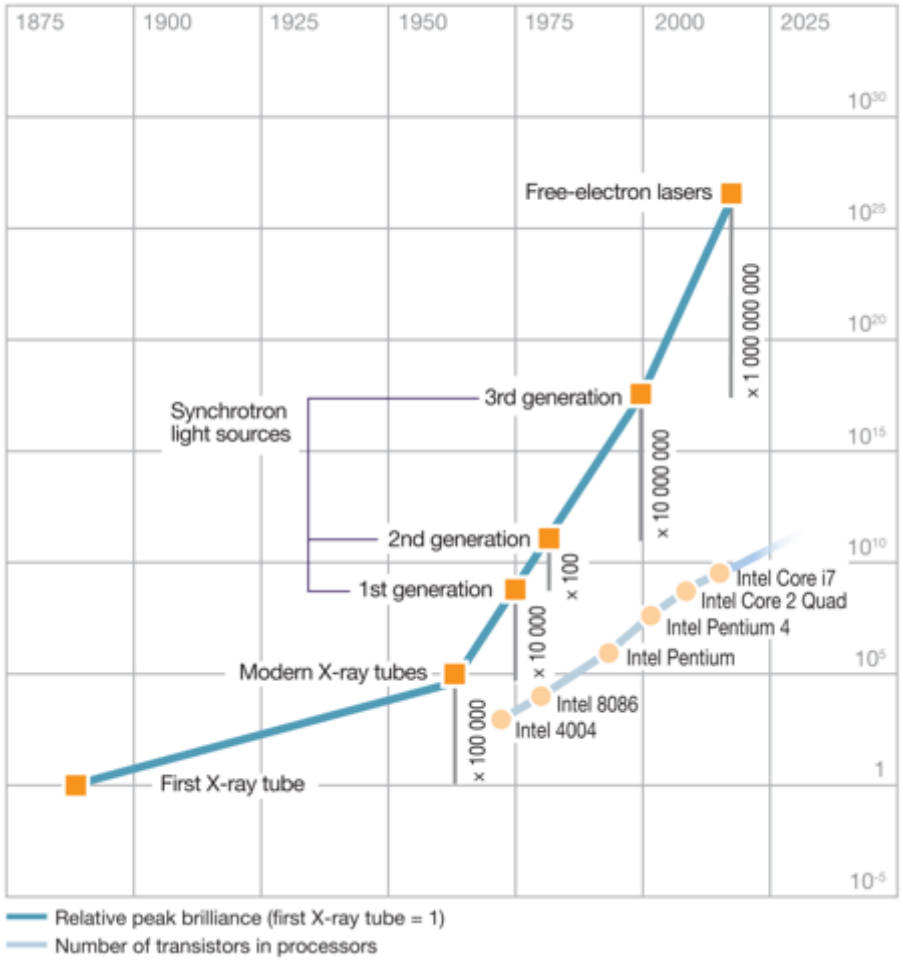
From Bytes/s to GBytes/s

Steffen Hauf, European XFEL GmbH
13.01.2020

Data Challenges at the European XFEL

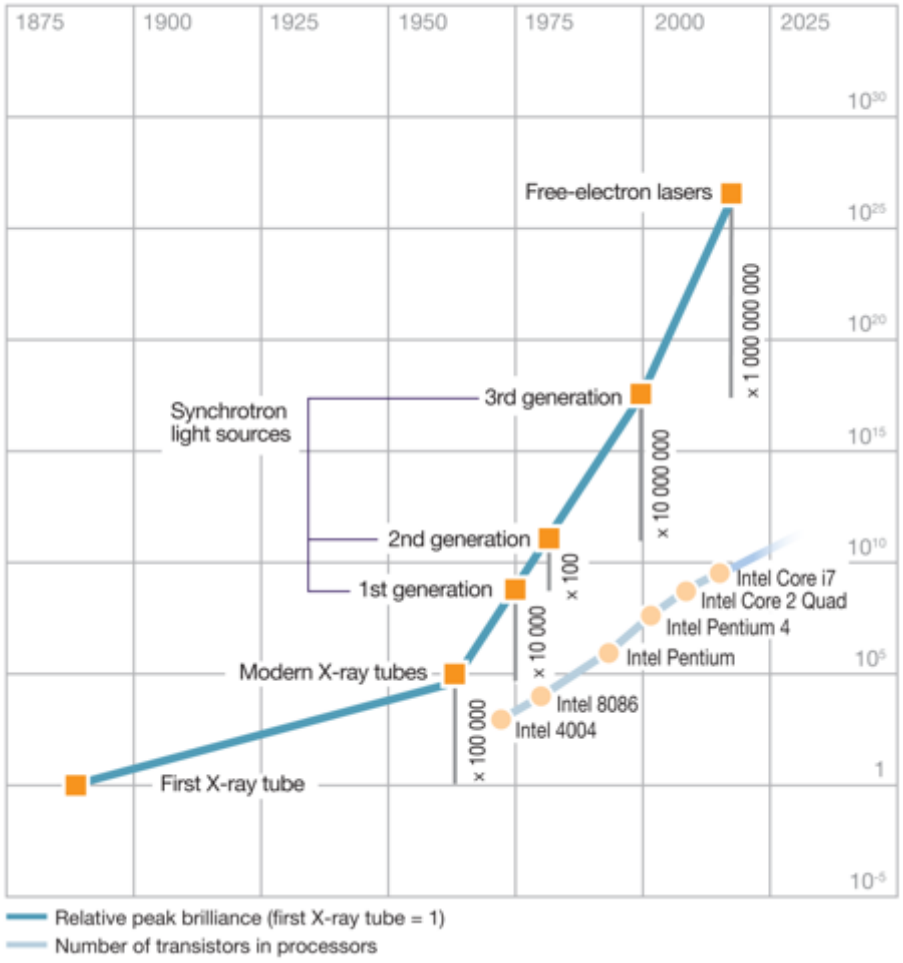
- The European XFEL – a (promotional) Introducion
- Data Challenges
- Data Drivers
- Data Solutions
- Lessons Learned and Outlook

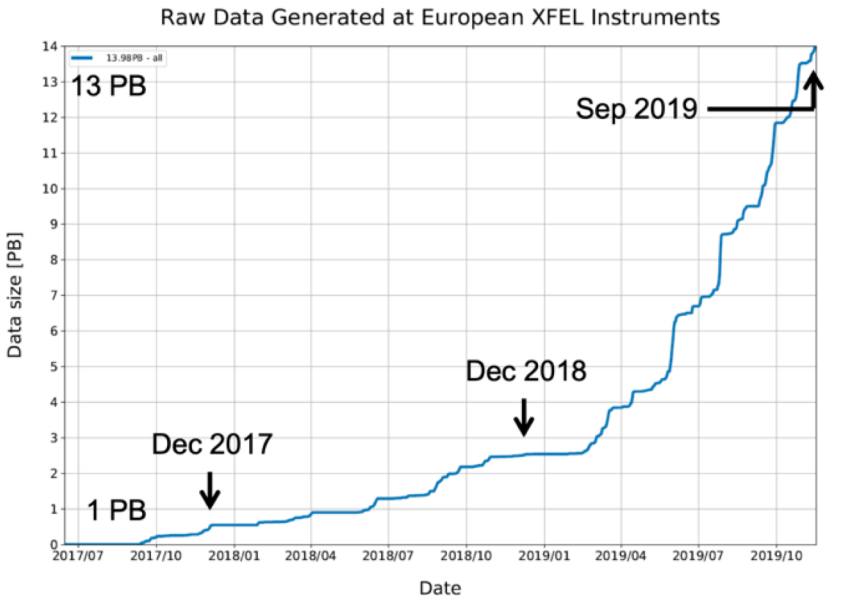
PR Slide: Light source development



- The development of light source facilities has been faster than the increase in computer processing capacity (i.e., Moore's Law)

PR Slide: Light source development





The European XFEL Facility

Schenefeld

Experiment hall

Laboratories

Offices

Osdorfer Born

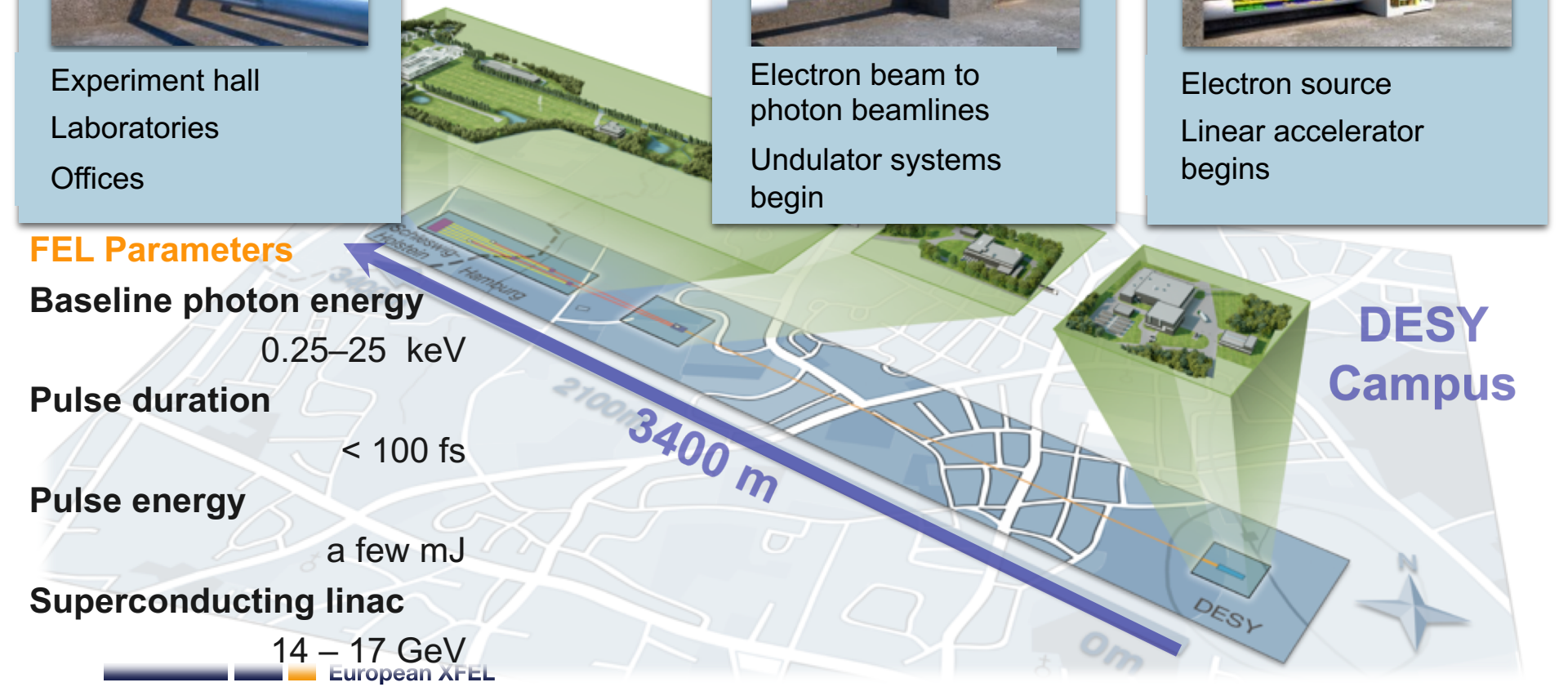
Electron beam to photon beamlines

Undulator systems begin

DESY-Bahrenfeld

Electron source

Linear accelerator begins



FEL Parameters

Baseline photon energy

0.25–25 keV

Pulse duration

< 100 fs

Pulse energy

a few mJ

Superconducting linac

14 – 17 GeV

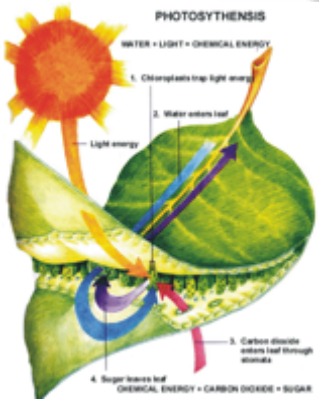
Key Scientific Goals

Science of condensed matter & materials

- ►Understand the cause of out-standing materials properties, in particular in dynamic behaviour
- ►Develop new & better materials

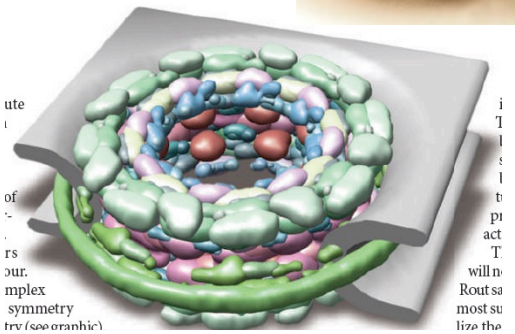
Ultrafast photochemistry & energy research

- ►‚Look' and understand how chemistry works on the atomic level
- ►Develop better chemistry



Structure of biomolecules, complexes & cells

- ►Determine function of biological systems through solving their 3D spatial structure
- ►Develop new treatments & drugs

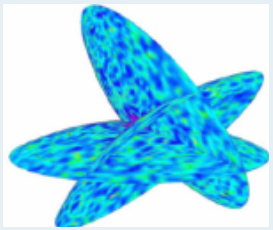


XFEL Scientific Instruments

Hard X-Rays

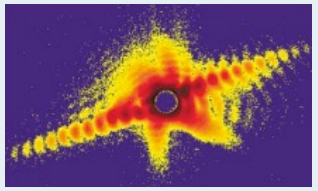
SPB **Single Particles, Clusters and Biomolecules and Serial Femtosecond Crystallography**

Will determine the structure of single particles, such as atomic clusters, viruses and biomolecules



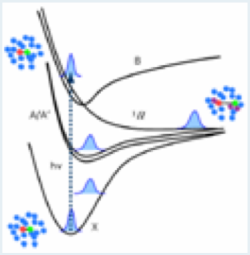
MID **Materials Imaging & Dynamics**

Will be able to image and analyse nano-sized devices and materials used in engineering



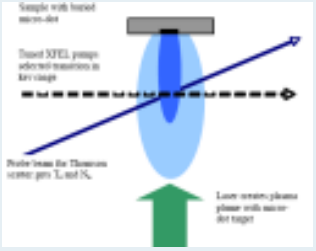
FXE **Femtosecond X-Ray Experiments**

Will investigate chemical reactions at the atomic scale in short time scales molecular movies



HED **High Energy Density Matter**

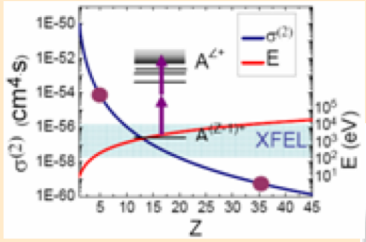
Will look into some of the most extreme states of matter in the universe, such as the conditions at the center of planets



Soft X-Rays

SQS **Small Quantum Systems**

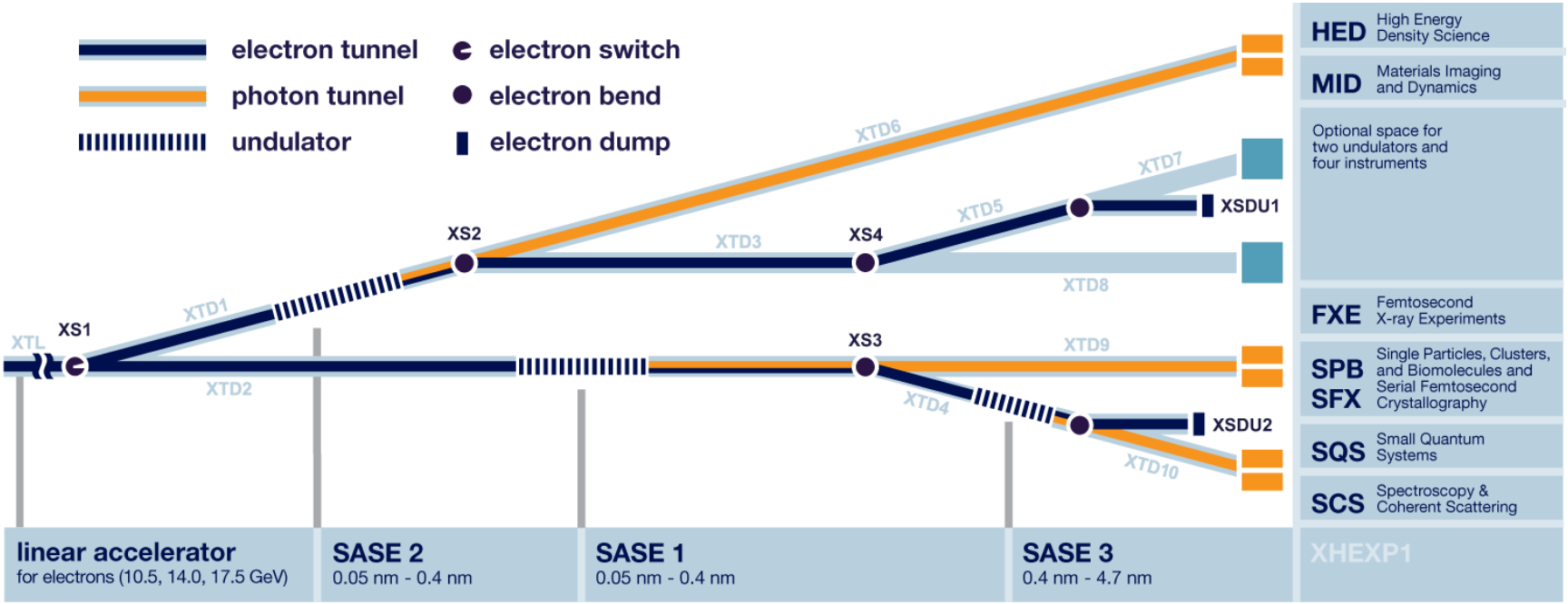
Will examine the quantum mechanical properties of atoms and molecules.



SCS **Soft X-Ray Coherent Scattering/Spectroscopy**

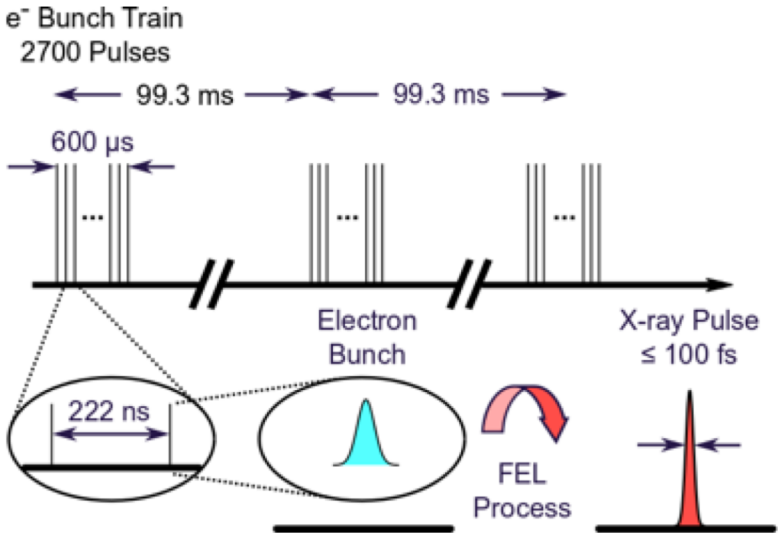
Will determine the structure and properties of large, complex molecules and nano-sized structures.

Beamline layout and experiment stations



European XFEL Time Structure

XFEL Burst Mode



The European XFEL pulse structure poses strict constraints on detectors (e.g. intensity and time structure)

Most of the time the use of commercial detectors is excluded

Most applications require 4.5 MHz repetition rate detectors

On average up to 27.000 pulses/s

Pulse duration

< 100 fs

High peak intensities

up to 10^{12} photons/pulse

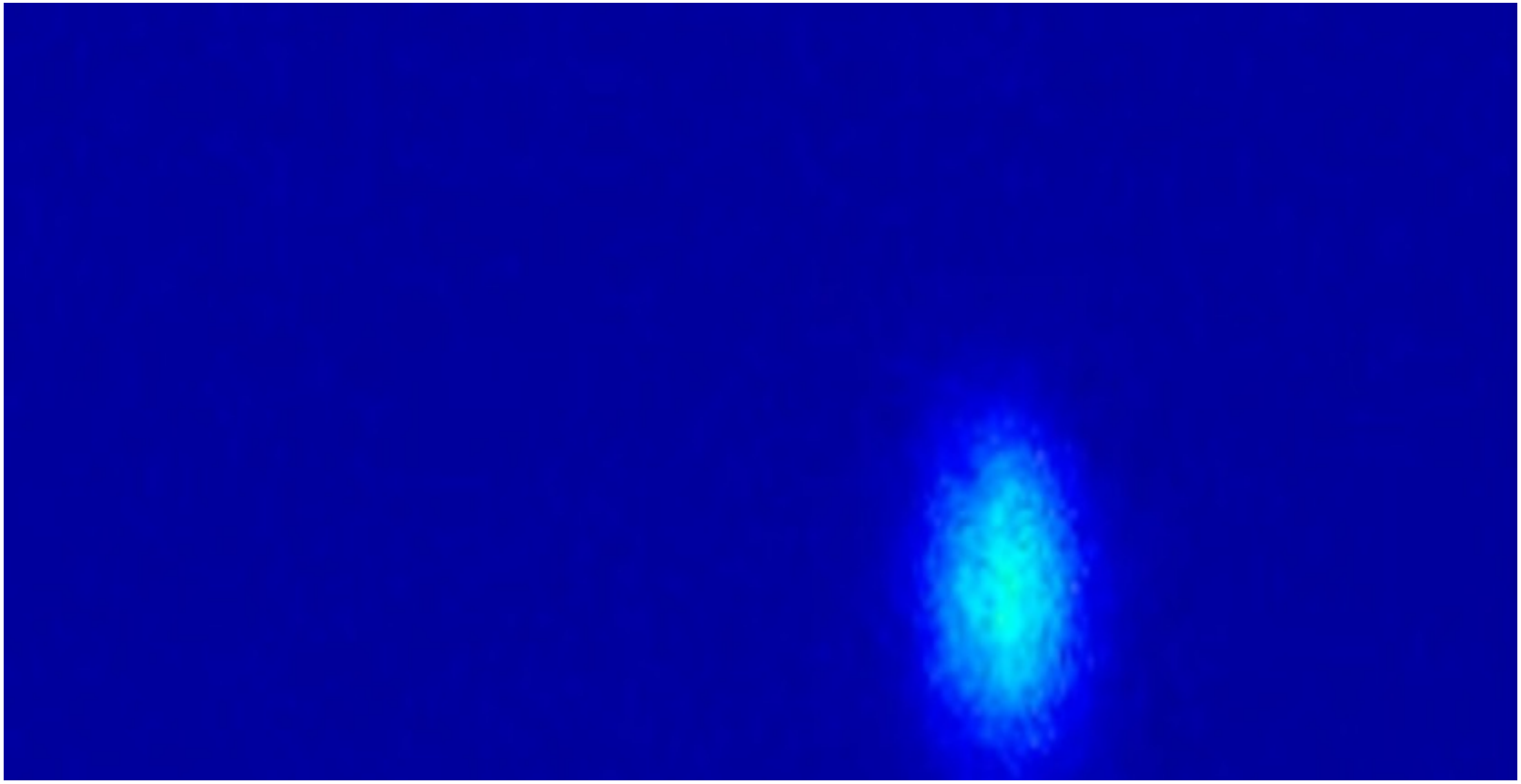
Various different pulse patterns possible

1 pulse per train

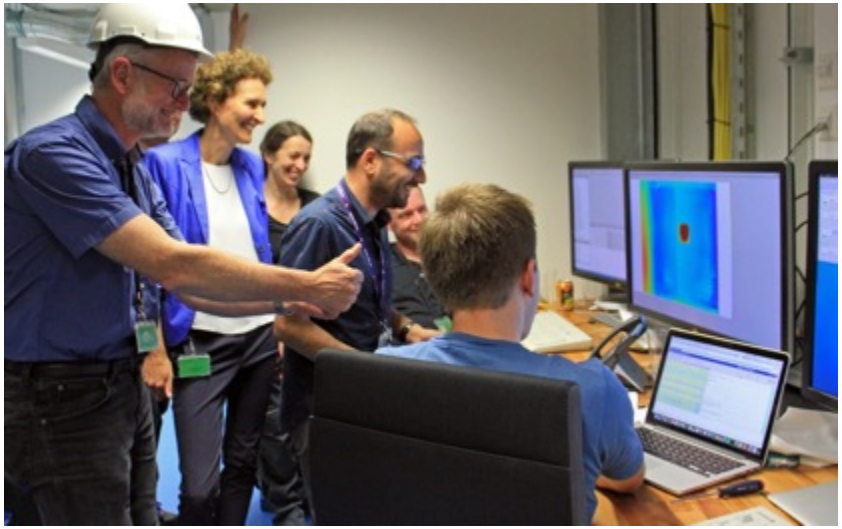
n pulses per train …

Linear, logarithmic or random distribution

4 May 2017—first lasing!



23 June 2017—first X-rays in the experiment hall!



1 September 2017—Inauguration and start of user operation

A strong laser from Elbphilharmonie greets European XFEL in the languges of the partner countries

Ribbon-cutting with high-ranking representatives of the partner countries

FXE and SPB/SFX instruments: available for users since Sept 2017

FXE instrument: ultrafast chemistry studies looking into catalysts, photosensitive materials, biochemistry

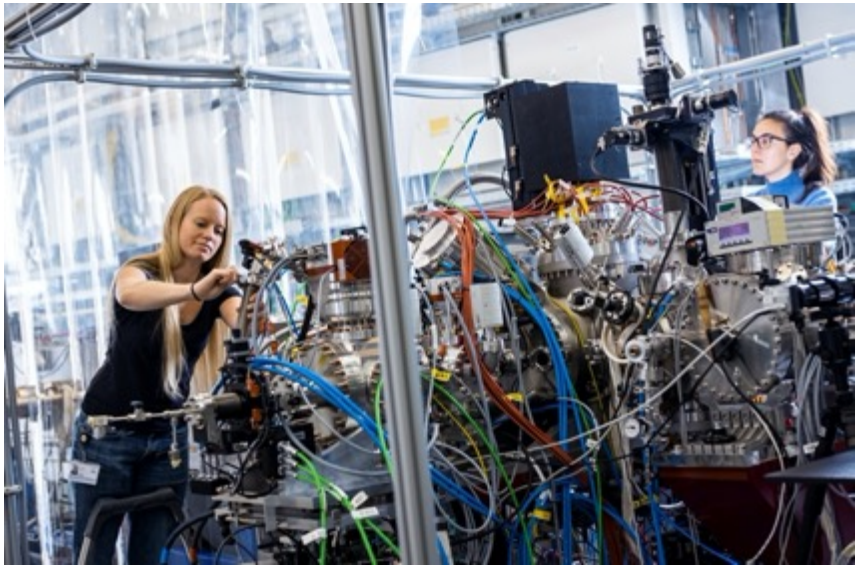
SPB/SFX instrument: structural biology, studies of atomic clusters, imaging of single cells, viruses, eventually molecules

FXE and SPB/SFX instruments: available for users since Sept 2017

FXE instrument: ultrafast chemistry studies looking into catalysts, photosensitive materials, biochemistry

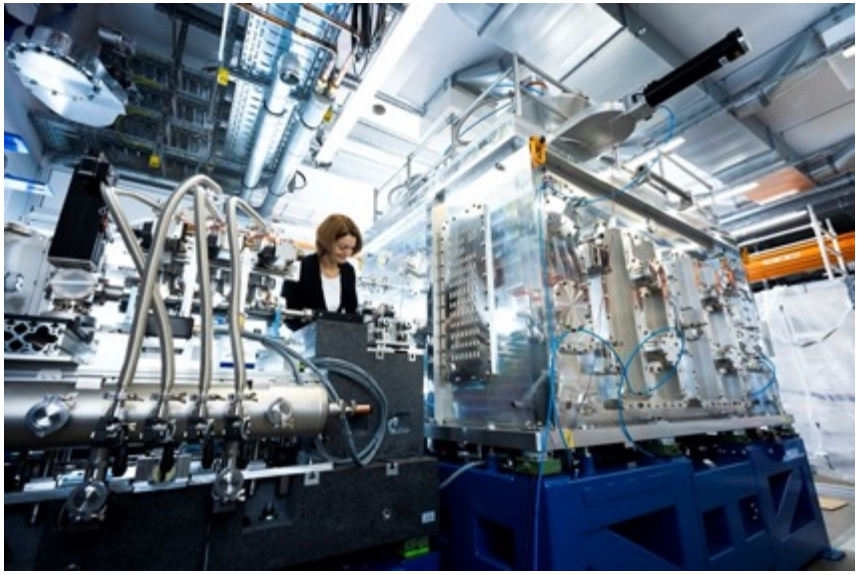
SPB/SFX instrument: structural biology, studies of atomic clusters, imaging of single cells, viruses, eventually molecules

SQS and SCS instruments: first users since late 2018

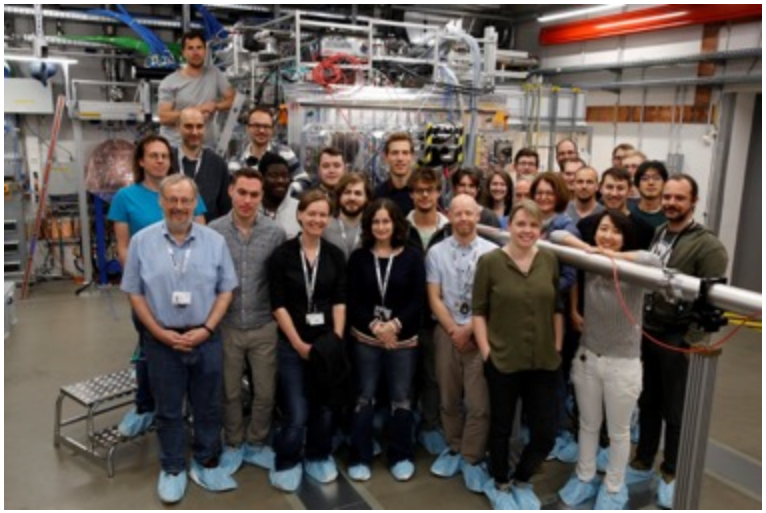


SQS and SCS instruments: first users since late 2018

HED and MID instruments: first users in 2019

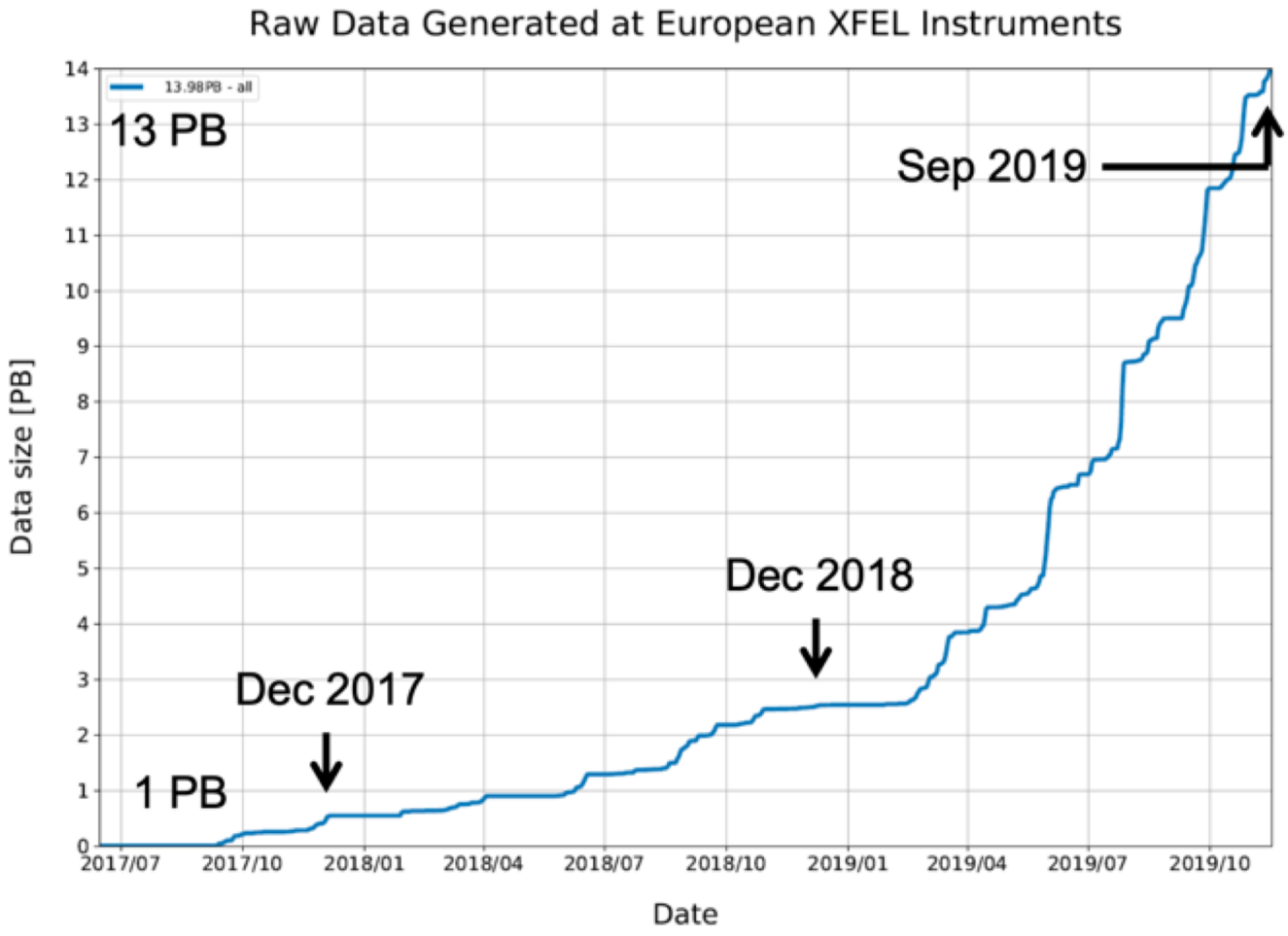


HED and MID instruments: first users in 2019



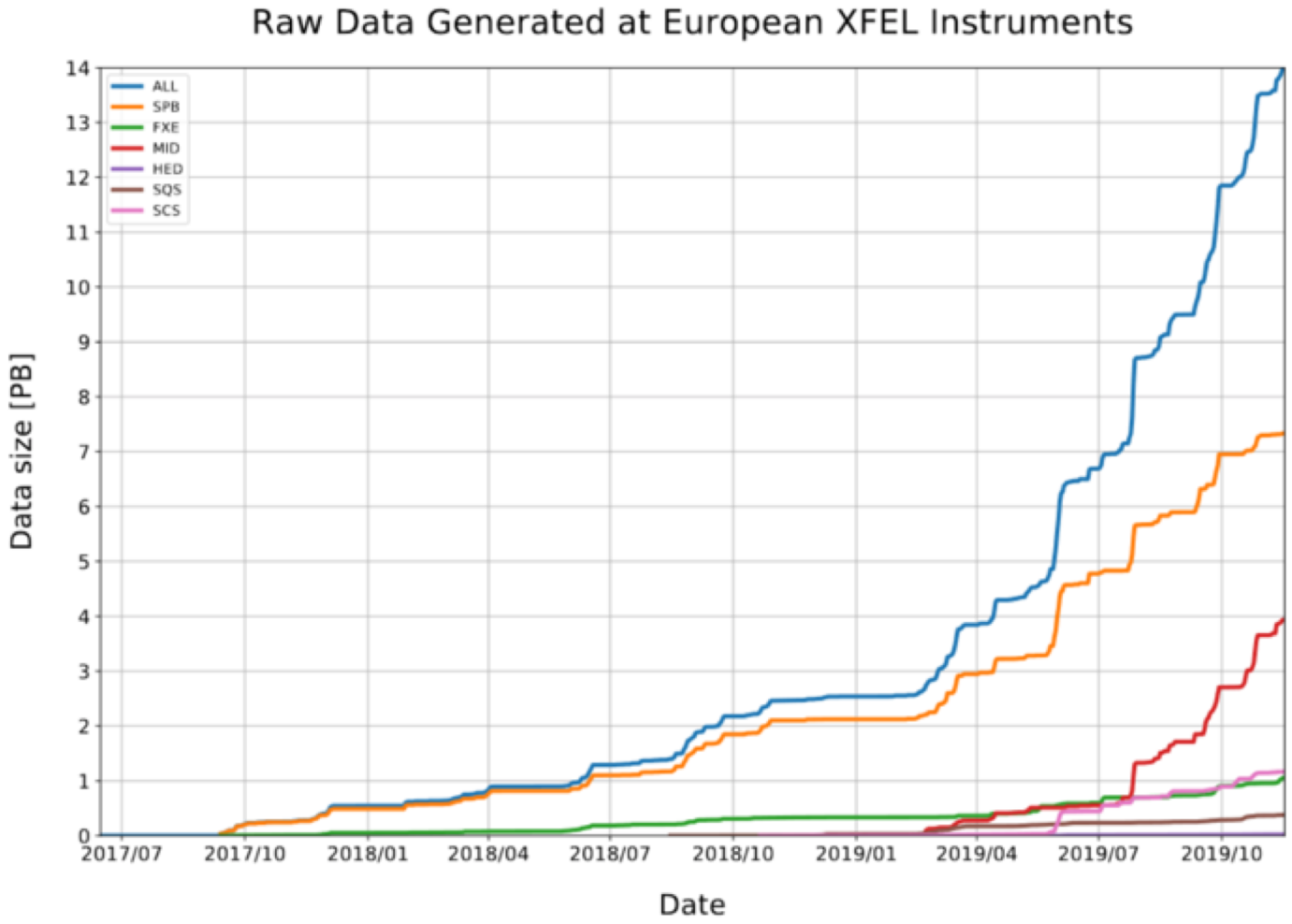
At MID no camera seems to have been around for the first experiment

The European XFEL is running and producing Data!

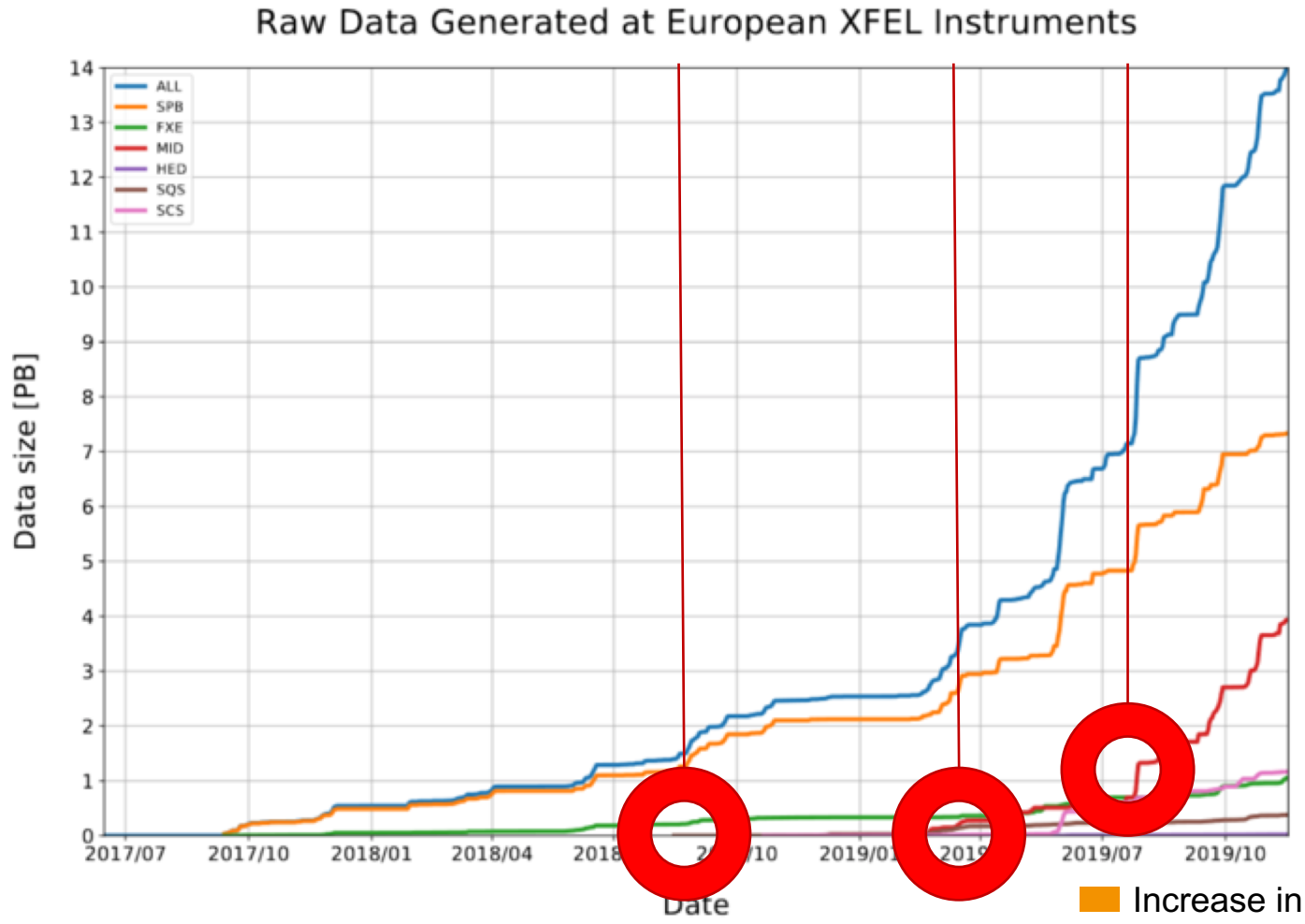


Data Challenges at the European XFEL

Data Challenges at the European XFEL

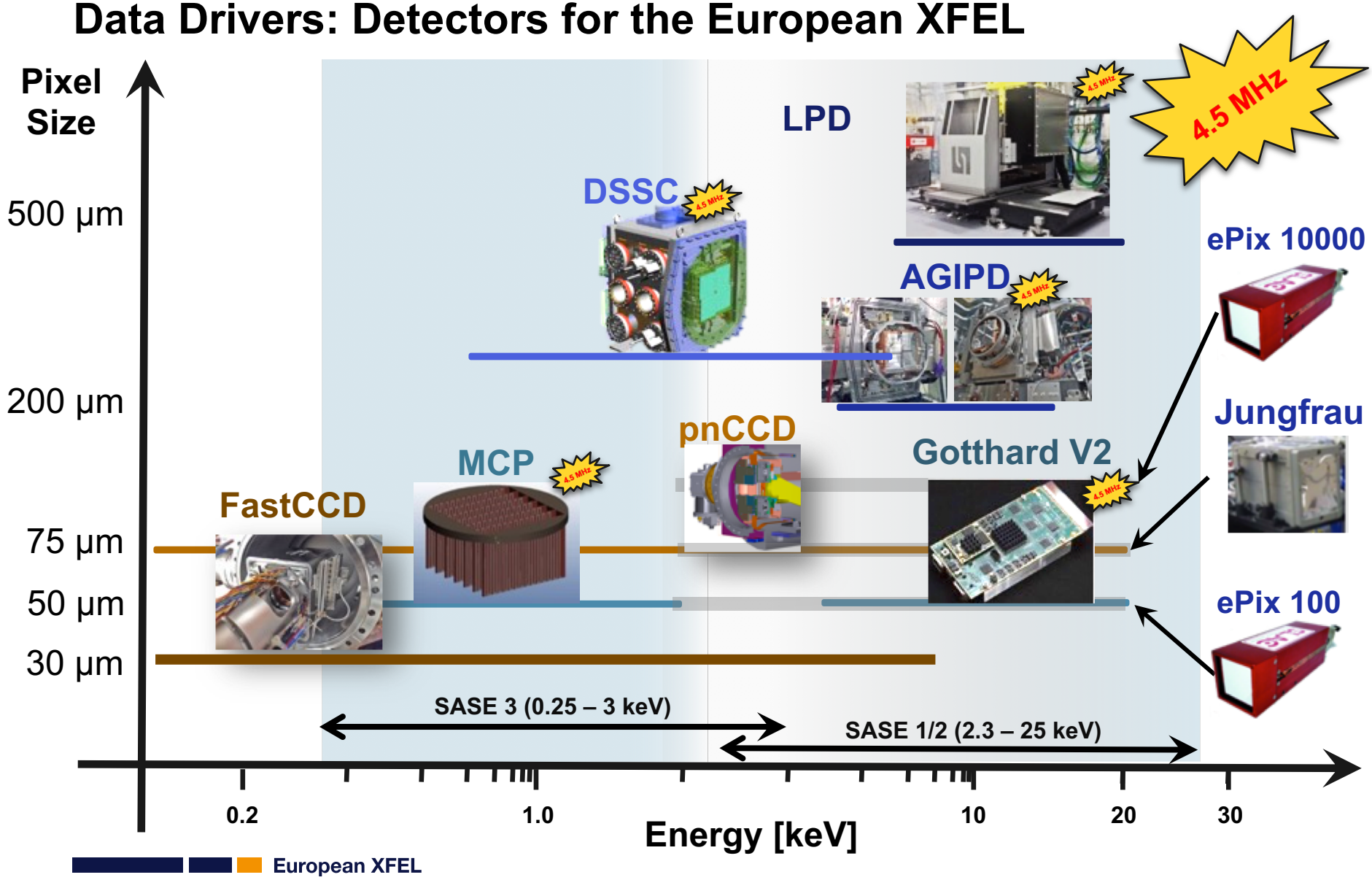


Data Challenges at the European XFEL



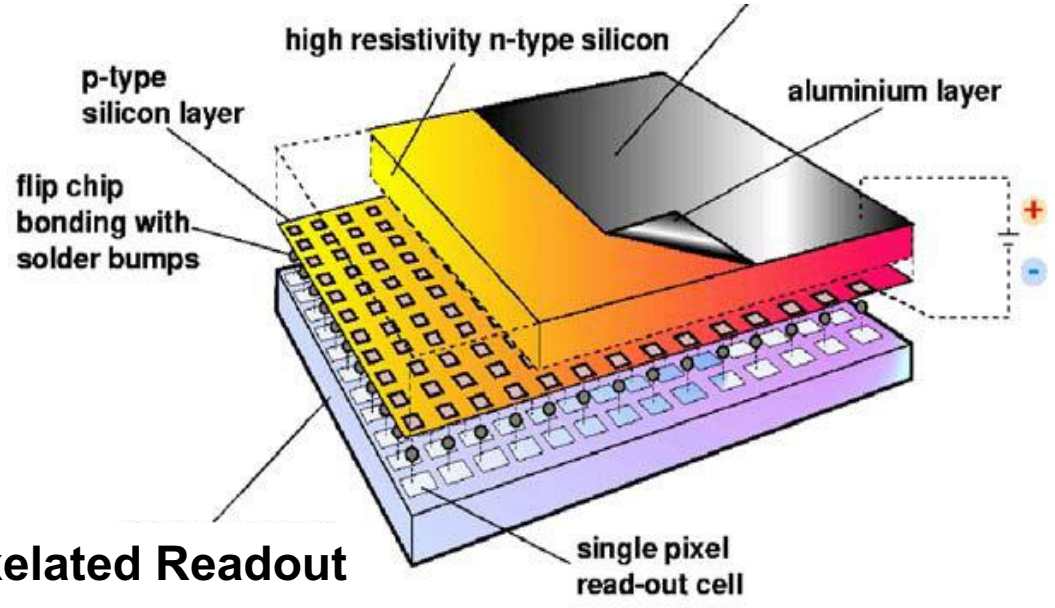
Data Challenges – Data Drivers

Data Drivers: Detectors for the European XFEL



Data Drivers: European XFEL Fast 2D Imagers – Hybrid Pixel Detectors

Pixelated Silicon Sensor

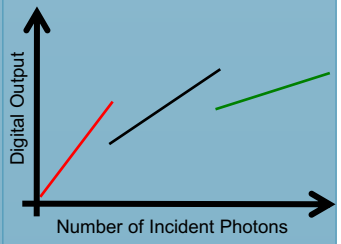
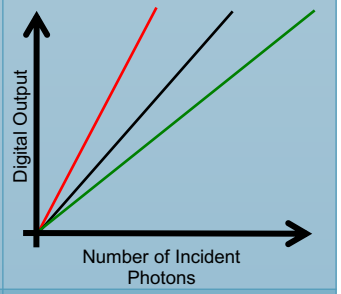
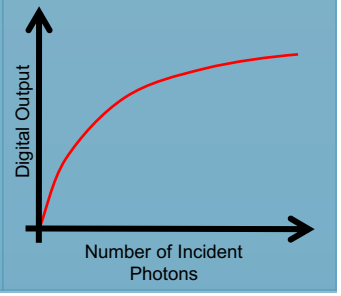


- Direct photon detection with Silicon sensor
  - High quantum efficiency
- Signal processing by read-out chip in each pixel
  - Amplification, AD conversion, storage in memory
- Fast read out up to several MHz and low power consumption
- Al entrance window
  - Optical/IR light blocking

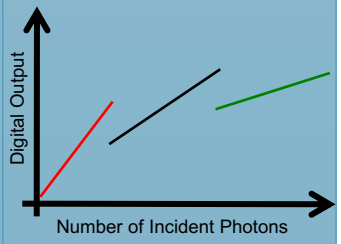
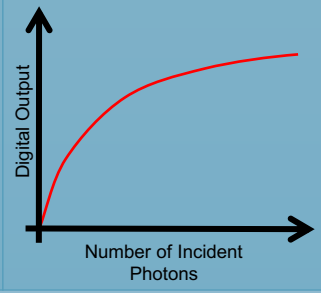
Pixelated Readout Chip (ASIC)

- Analog or digital memory
  - Capacity up to 800 cells/pixel
- Veto and trigger capability
  - Overwrite empty images

Data Drivers: 4.5 MHz High Dynamic Range Imaging Detectors

Detector	Specs	Modularity	Gain	Gain Curve
AGIPD 	1 Mpixel, 4.5 MHz 352 memory cells 200 µm sq. pixels 1-10^4 ph@ 12 keV 3 – 13 keV	**16 modules** in 2 cols x 8 rows on 4 quadrants	3 gains with automatic switching	
LPD 	1 Mpixel, 4.5 MHz 510 memory cells 500 µm sq. pixels 1-10^5 ph@ 12 keV 5 – 25 keV	16 modules per Super Module (2x8) **16 SM** on 4 quadrants	3 parallel gain stages with on front-end selection	
DSSC 	1 Mpixel, 4.5 MHz 800 memory cells ≈236 µm hex. pixels 1-10^4 ph@ >1keV 0.5 – 6 keV	**16 modules** in 2 cols x 8 rows on 4 quadrants	Linear gain in ASIC (miniSDD) non-linear gain in sensor (DePFET)	

Data Drivers: 4.5 MHz High Dynamic Range Imaging Detectors

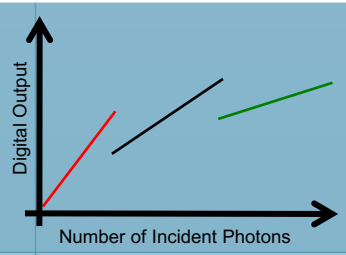
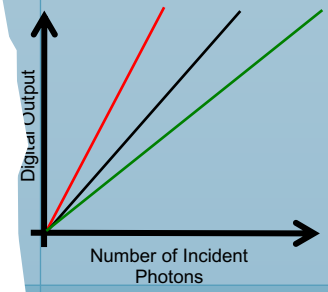
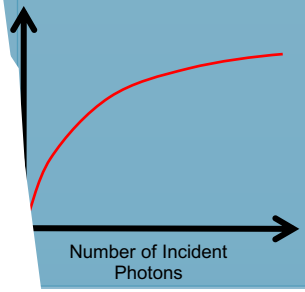
Detector	Specs	Modularity	Gain	Gain Curve
AGIPD 	1 Mpixel, 4.5 MHz 3 – 13 keV	**16 modules**	3 gains with ching	
LPD 	1 Mpixel, 4.5 MHz	16 modules per	3 parallel gain stages with on front-	
DSSC 	1 Mpixel, 4.5 MHz	**16 modules** in 2	Linear gain in ASIC	

Project Leader: H. Graafsma, DESY
PSI/SLS Villingen, University Bonn, University Hamburg, DESY

Project Leader: M. Hart, RAL/STFC
Rutherford Appleton Laboratory/STFC
University of Glasgow

Project Leader: M. Porro, European XFEL
University Heidelberg, Politecnico di Milano, Università di Bergamo, DESY, European XFEL

Data Drivers: 4.5 MHz High Dynamic Range Imaging Detectors

Detector	Specs	Gain Curve
AGIPD 	1 Mpixel, 4.5 352 memory 200 µm sq. pi 1-10^4 ph@ 12 3 – 13 keV	
LPD 	1 Mpixel, 4.5 M 510 memory ce 500 µm sq. pixe 1-10^5 ph@ 12 k 5 – 25 keV	
DSSC 	1 Mpixel, 4.5 MH 800 memory cell ≈236 µm hex. pi 1-10^4 ph@ >1ke 0.5 – 6 keV	

- Three gains per pixels
- Analogue gain evaluation
- Analogue memory
- ….

- Six gains per pixels
- Analogue memory
- Air scattering in FF
- ….

- Non-linear gain
- One ADU per photon
- ....

Data Drivers: 4.5 MHz High Dynamic Range Imaging Detectors

Detector	Specs
AGIPD 	1 Mpixe 352 me 200 µm 1-10^4 p 3 – 13 k
LPD 	1 Mpixe 510 me 500 µm 1-10^5 ph@ 12 k 5 – 25
DSSC 	1 Mp 800 ≈236 1-10 0.5 -

Example LPD

x 512 memory cells
x 1 million pixel
= 5 x 10^8 parameters
and
3 Gain Stages
2 Gain Settings

~ 10^9 Parameters

- Three gains per pixels
- Analogue gain evaluation
- Analogue memory
- ….

Parameter Dependence

Temperature,
integration time,
irradiated dose,
bias voltage, detector
configuration and …

- Six gains per pixels
- Analogue memory
- Air scattering in FF
- ….

- Non-linear gain
- One ADU per photon
- ....

Data Drivers: Detectors for the Scientific Instruments

SASE I / High E	Single Particles, Clusters and Biomolecules (SPB)	AGIPD	Gotthard V1/2			Jungfrau
SASE II / High E	Materials Imaging & Dynamics (MID)	AGIPD	Gotthard V1/2		ePix	Jungfrau
SASE II / High E	Femtosecond X-ray Experiments (FXE)	LPD	Gotthard V1/2			Jungfrau
SASE II / High E	High Energy Density Matter (HED)	Jungfrau	Gotthard V1/2		ePix	Jungfrau
SASE III / Low E	Small Quantum Systems (SQS)	DSSC		pnCCD		MCP
SASE III / Low E	Spectroscopy and Coherent Scattering (SCS)	DSSC	Fast CCD			MCP

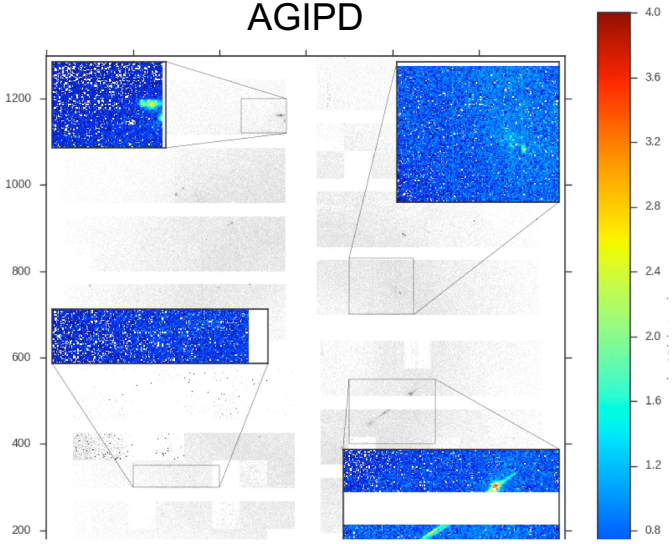
Data Drivers: Detectors for the Scientific Instruments

SASE I / SASE II — High E	Single Particles, Clusters and Biomolecules (SPB)	AGIPD, Gotthard V1/2, Jungfrau
SASE I / SASE II — High E	Materials Imaging & Dynamics (MID)	AGIPD, Gotthard V1/2, ePix, Jungfrau
SASE I / SASE II — High E	Femtosecond X-ray Experiments (FXE)	LPD, Gotthard V1/2, Jungfrau
SASE I / SASE II — High E	High Energy Density Matter (HED)	Jungfrau, Gotthard V1/2, ePix, Jungfrau
SASE III — Low E	Small Quantum Systems (SQS)	DSSC, MCP, pnCCD
SASE III — Low E	Spectroscopy and Coherent Scattering (SCS)	DSSC, Fast CCD

Data Drivers: Data Examples from Detectors

- Data comes in a variety of „flavors“
- Not trivial to generically reduce, as very experimental context dependent
- Usefulness of data determined by calibration quality

AGIPD



LPD

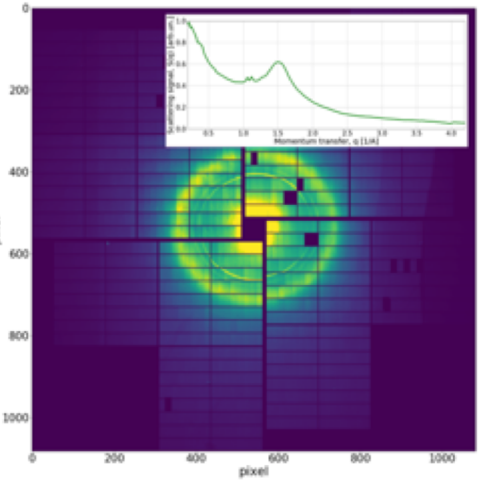
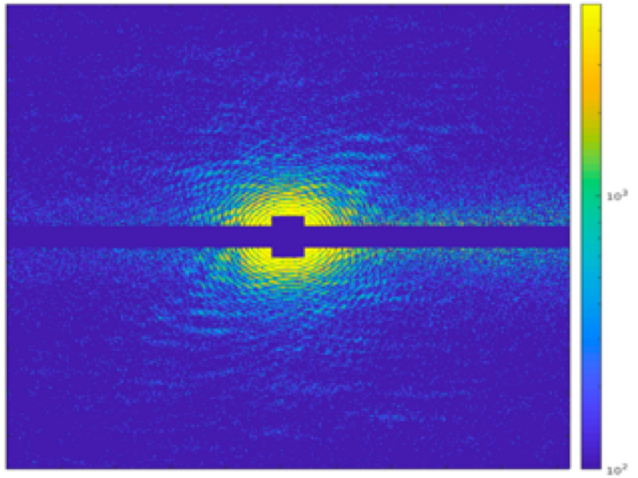
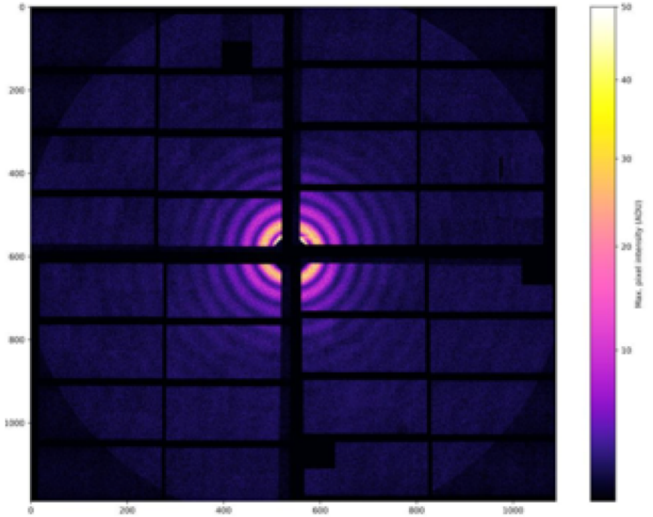


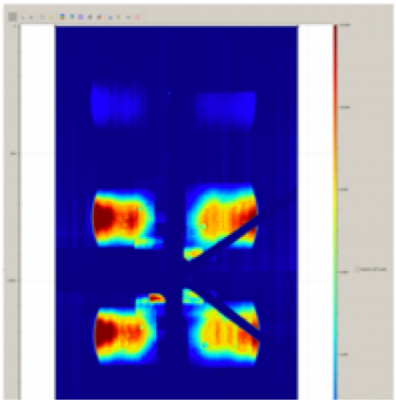
Figure 8: Liquid scattering pattern of tetrahydrofuran solution of a Cu complex collected with the LPD detector [23,24] at scientific instrument FXE [25](corrected for dark offset). Inset: Average of the azimuthally integrated set of 150 image.

Doped He nanodroplets imaging
PI: R. Tanyag, D. Rupp (TU/MBI Berlin)





FastCCD



Data Drivers: Digitizers & FPGAs

MicroTCA Crates

Large 12 slot 9U and small 6 slot 2U (including MCH, Power Supply and CPU)

X2Timer

XFEL Timing System module for synchronization (clocks and triggers) and pulse parameters from NAT

DAMC2

Required for Clock & Control system for fast 2D detectors, VETO System, Machine Protection System and photon beam loss monitors from DESY

SIS8300

Fast 125MSPS ADC with 10 channels and 16bit resolution for diagnostics and detectors from Struck Innovative Systeme

ADQ412/ADQ14/ADQ7

High-speed digitizers from 1.8GSPS to 10GSPS with 12 to 14 bit resolution from Teledyne SP Devices

Data Drivers: Digitizers & FPGAs

MicroTCA Crates

Large 12 slot 9U and small 6 slot 2U (including MCH, Power Supply and CPU)

X2Timer

XFEL Timing System module for synchronization (clocks and triggers) and pulse parameters from NAT

DAMC2

Required for Clock & Control system for fast 2D detectors, VETO System, Machine Protection System and photon beam loss monitors from DESY

SIS8300

Fast 125MSPS ADC with 10 chan els and 16bit resolu diagnostics detec Innovati

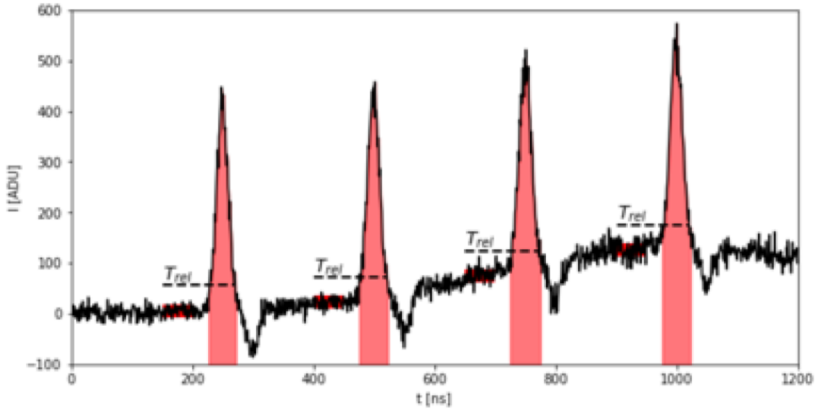
ADQ412/ DQ14/ADQ7

High- d digitizers GSPS to 2 to 14 ces

1 MB/s - 1 GB/s
10 Hz burst

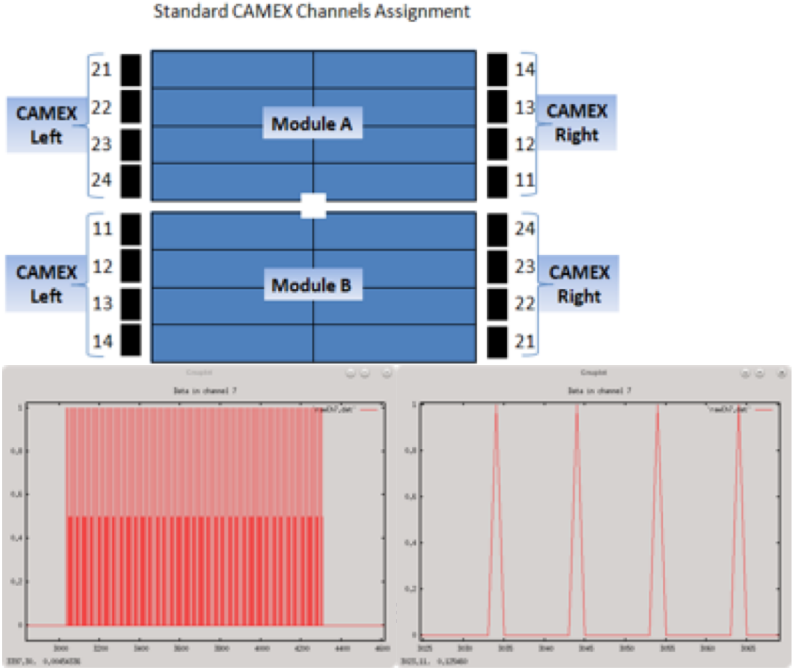
Data Drivers: Data Examples from Digitizers & FPGAs

- Raw data to very condensed derived data
- Processing on FPGA
- Processing in software
- Often used in diagnostics
  → data features more fixed than detectors recording X-rays from user sample



ROI sampling, e.g. from MCP data

pnCCD sampling at pixel clock



Joakim Laksman *et al.* • EuXFEL photoelectron spectrometer

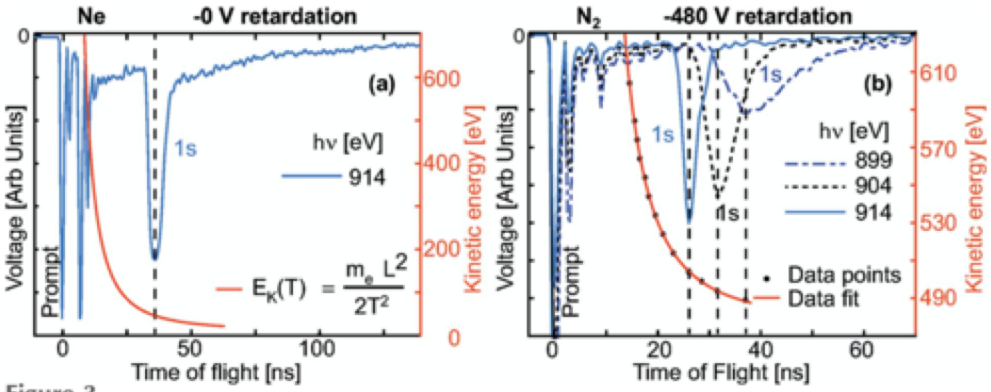
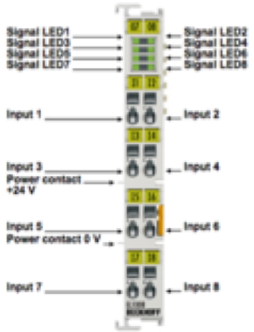


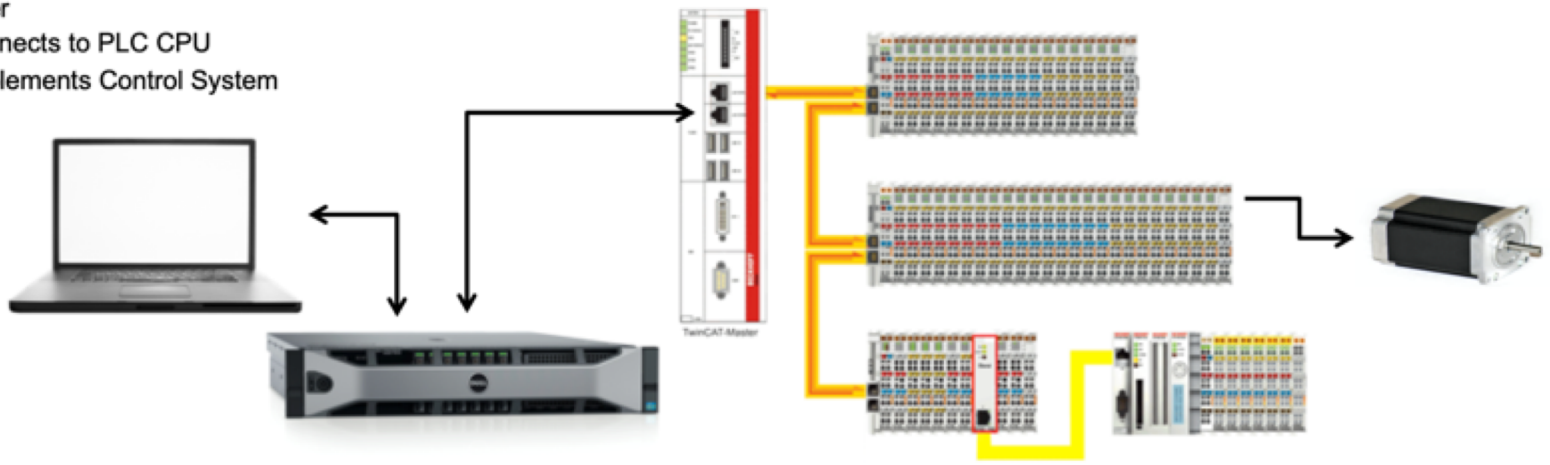
Figure 3
(*a*) 0 V retardation. Electron TOF spectrum after Ne 1*s* ionization at 914 eV. TOF is 36.25 ns which corresponds to a kinetic energy of 44.2 eV according to equation (1) (red curve). (*b*) −480 V retardation. Electron TOF spectrum after N_2 1*s* ionization at different photon energies. Peak center (black dots) fitted to equation (2) (red curve). Spectra are averaged over 100 pulses for higher statistics.

Data Drivers: PLCs and other “slow“ Data

- **P**rogrammable **L**ogic **C**ontroller
- Terminals as interface to h/w
- Terminals are connected together
- PLC CPU
  - Connects via cables to Terminals
  - Implements programs for control
- Computer
  - connects to PLC CPU
  - Implements Control System

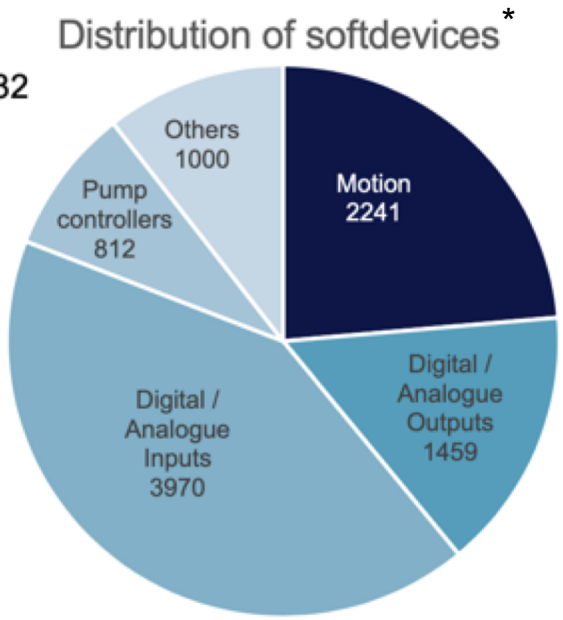


Terminals (i.e. digital input)



Data Drivers: PLCs and other “slow“ Data

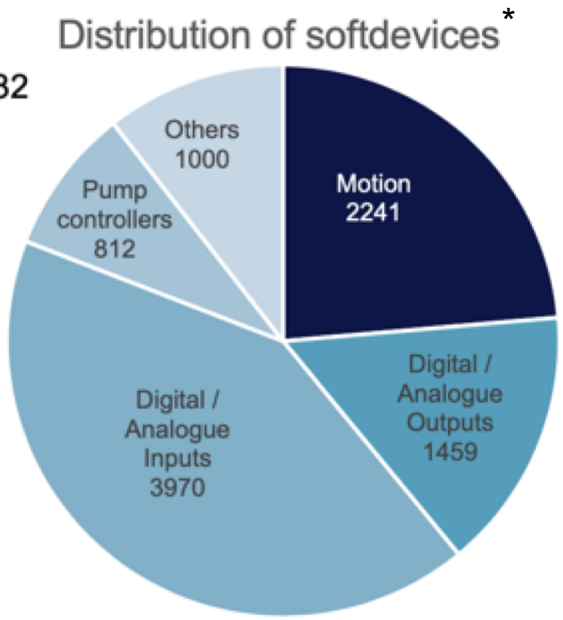
- Softdevices/Equipment: 9.482
- Number of Terminals: 8.452
- PLC Loops / CPUs: 120
- PLC Modules / crates: >500
- MicroTCA Systems: 35
- Digitizer channels: >280



*Status Sept. 2019

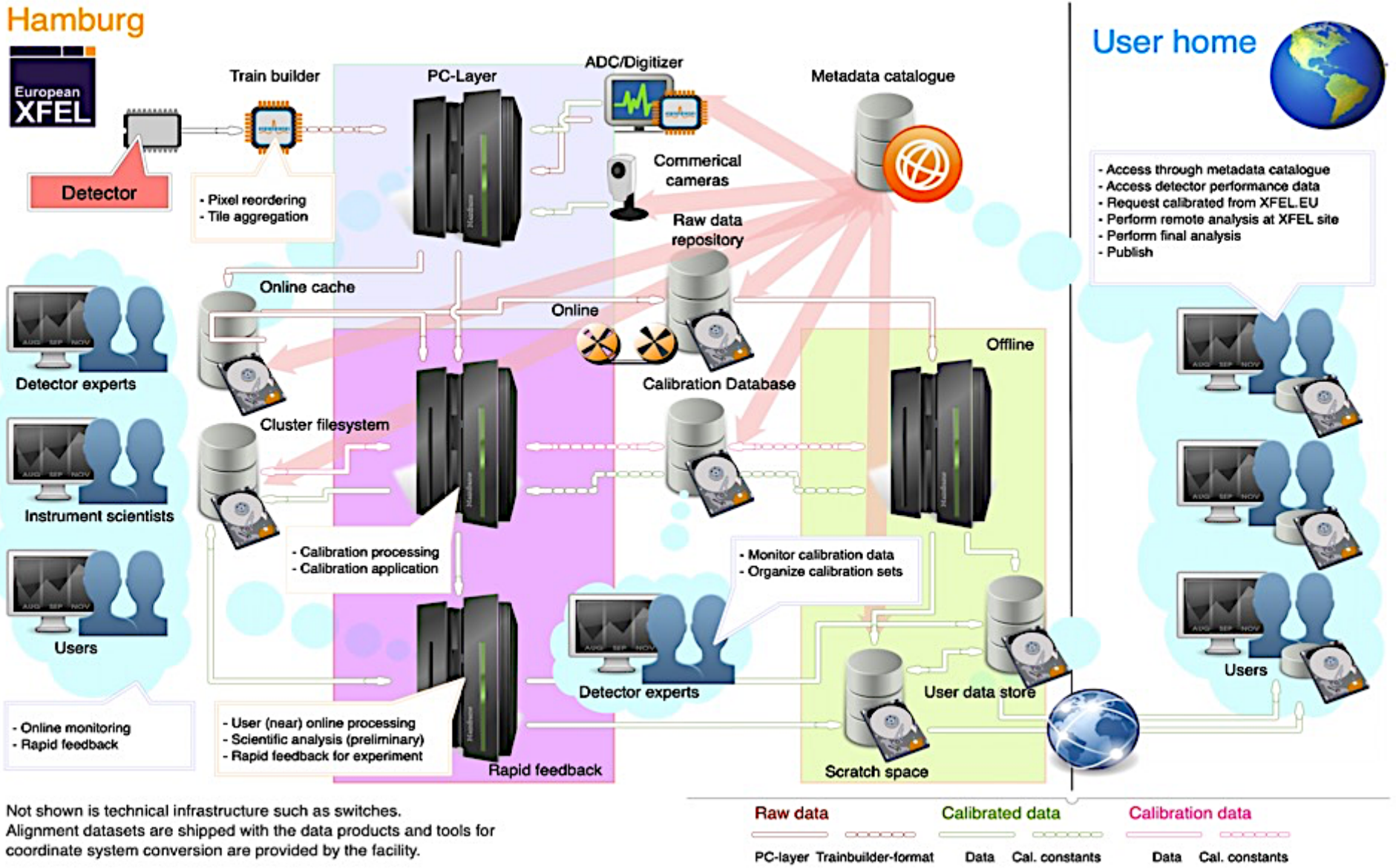
Data Drivers: PLCs and other “slow“ Data

- Softdevices/Equipment: 9.482
- Number of Terminals: 8.452
- PLC Loops / CPUs: 120
- PLC Modules / crates: >500
- MicroTCA Systems: 35
- Digitizer channels: >280



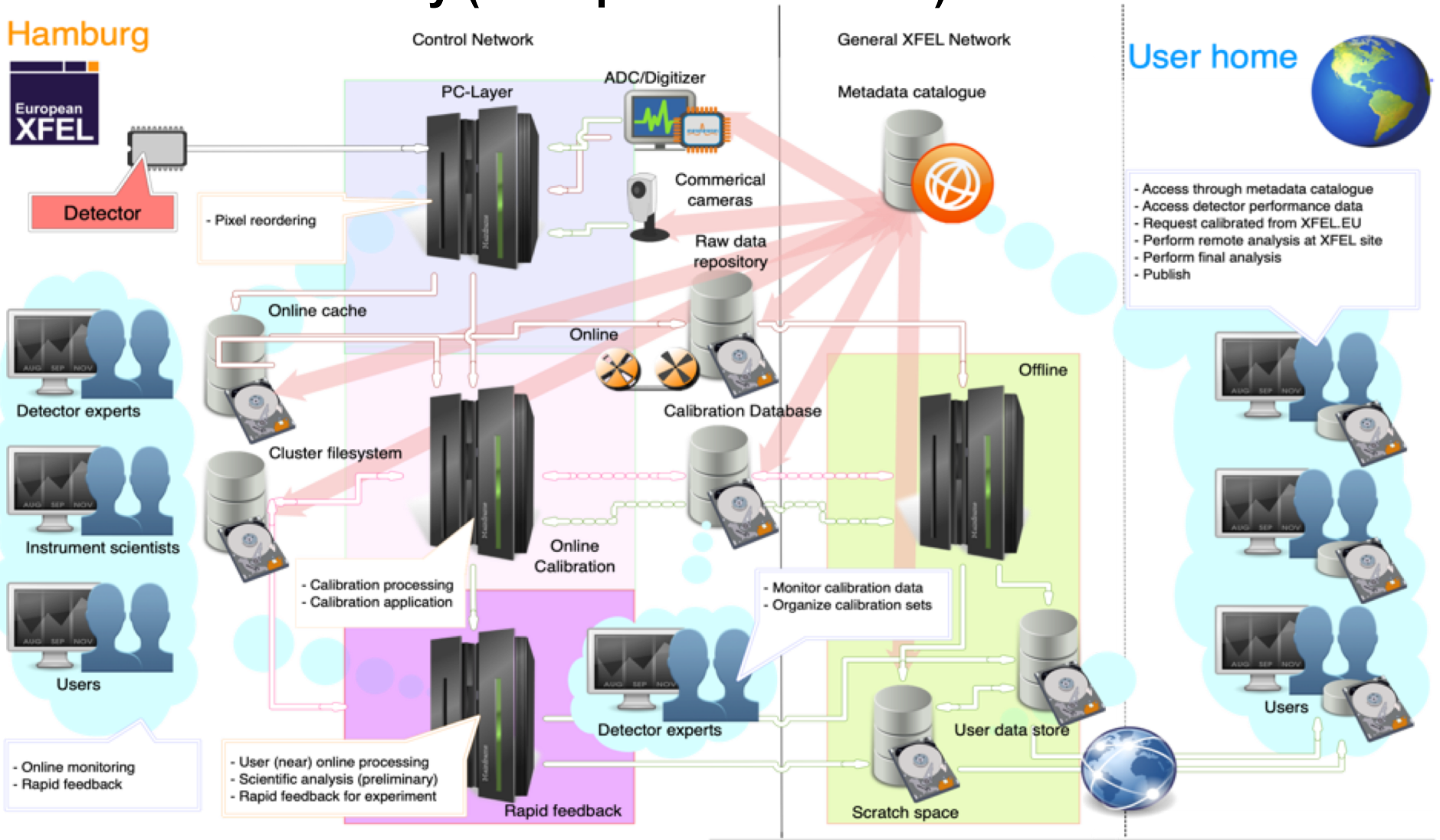
*Status Sept. 2019

Data Flow Concept (as planned in 2015)



Not shown is technical infrastructure such as switches.
Alignment datasets are shipped with the data products and tools for coordinate system conversion are provided by the facility.

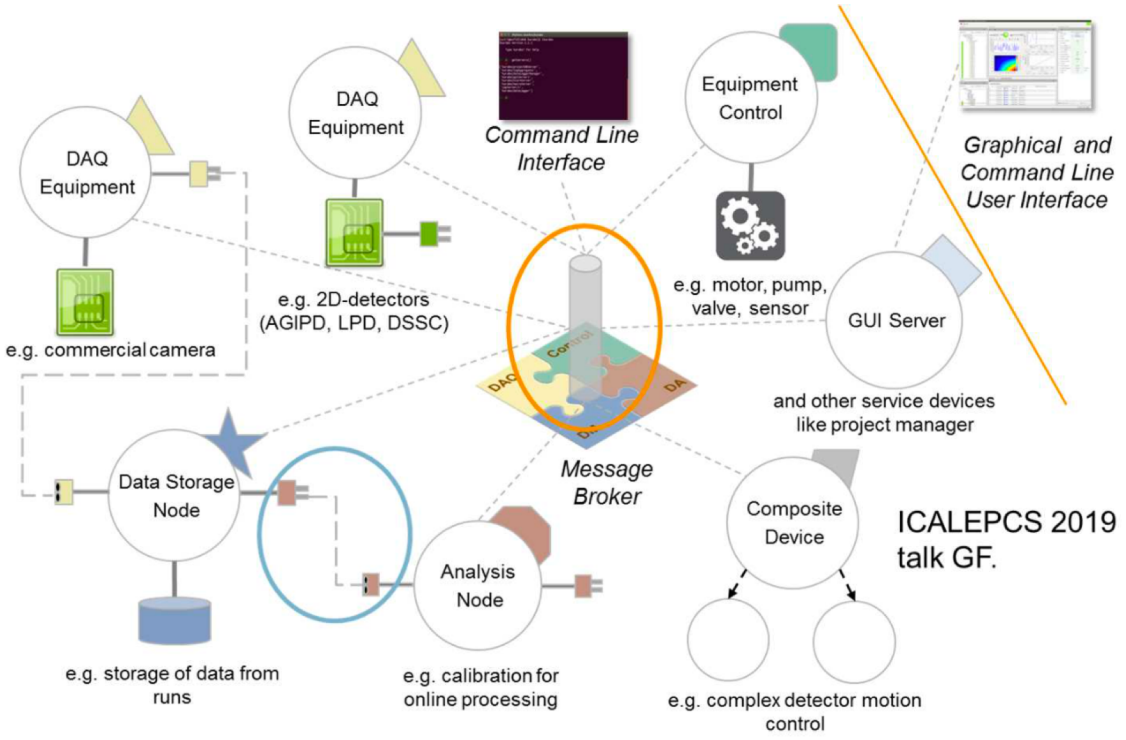
Data Flow Reality (as implemented 2019)



Not shown is technical infrastructure such as switches.
Alignment datasets are shipped with the data products and tools for coordinate system conversion are provided by the facility.

Data Drivers: Karabo, the XFEL.EU SCADA Framework

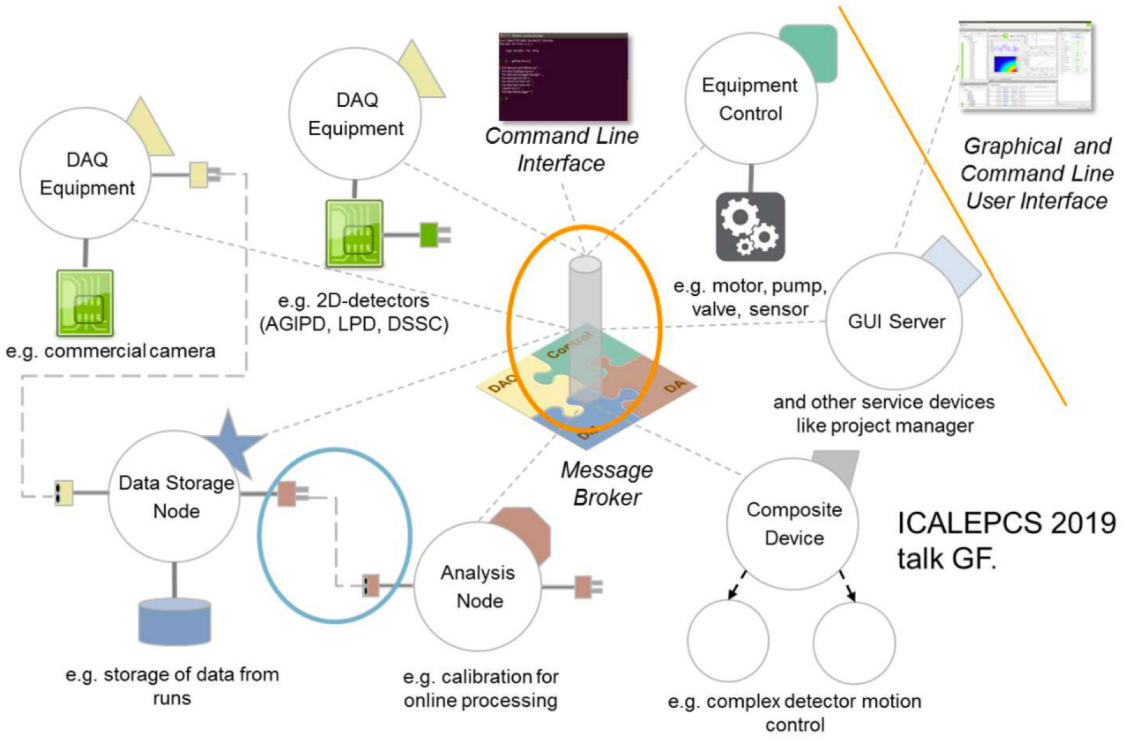
- **Communication**
  - Broker: cmds, values, schemas
    - ► Partitioning: topics SA1, SA2, SPB…
  - Data: p2p TCP
- **Apis**
  - cpp (c++, boost)
  - bound (Py 3.6 bound on cpp)
  - middlelayer (Py 3.6)
- **Devices – everything's a device**
  - reflect h/w (tight for Beckhoff)
  - control few to many other devices
  - interface to other services
- **Device servers – run devices**
  - Multi-thread + event loop + GIL handling
- **User interfaces**
  - Gui-client
  - CLI



- Photon systems + Experiments: Karabo
- Accelerator: DOOCs
- Any some EPIX, TINE, Tango niches

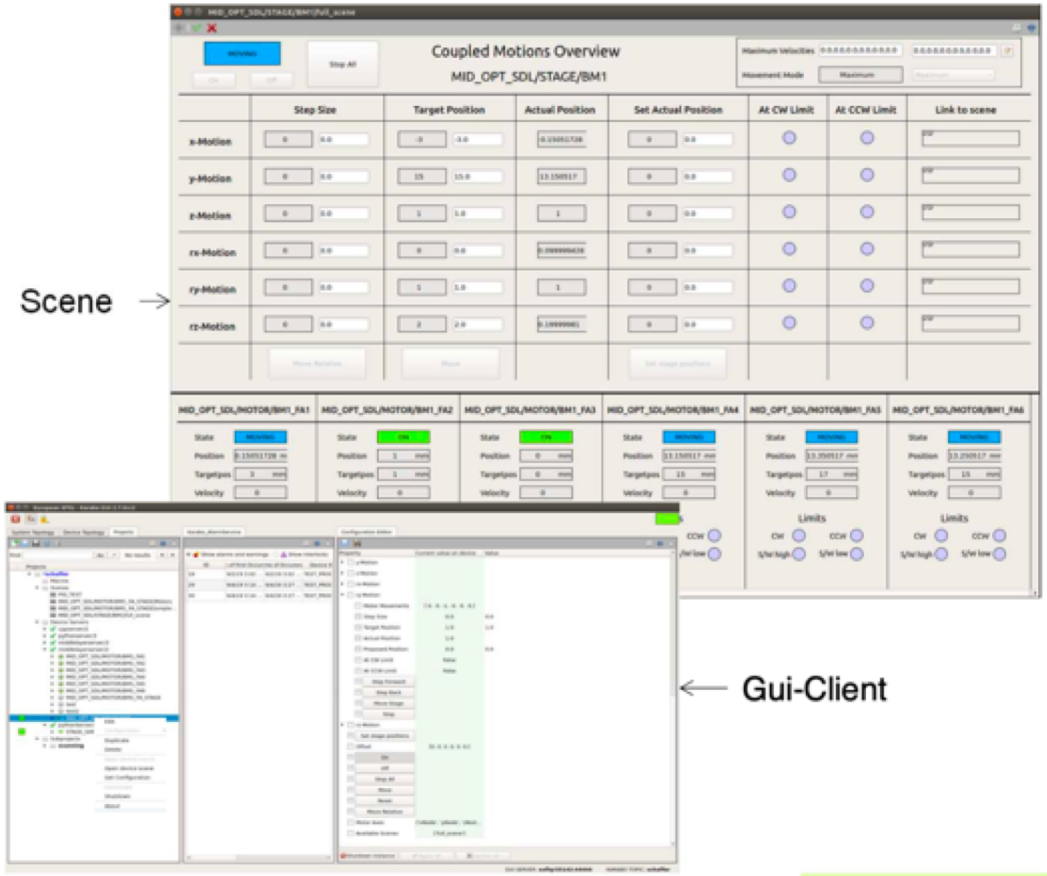
Data Drivers: Karabo, the XFEL.EU SCADA Framework

- **Communication**
  - Broker: cmds, values, schemas
    - ► Partitioning: topics SA1, SA2, SPB…
  - Data: p2p TCP
- **Apis**
  - cpp (c++, boost)
  - bound (Py 3.6 bound on cpp)
  - middlelayer (Py 3.6)
- **Devices – everything's a device**
  - reflect h/w (tight for Beckhoff)
  - control few to many other devices
  - interface to other services
- **Device servers – run devices**
  - Multi-thread + event loop + GIL handling
- **User interfaces**
  - Gui-client
  - CLI



- SPB Instrument: 104 device servers and 1452 devices on 28 ITDM control servers
- Facility total (Sept. 2019): ~14000 devices, 1.7 million control parameters
- **~30 GB/day → soon to move data logging to influxDB + Grafana access**

Data Drivers: Karabo, the XFEL.EU SCADA Framework



- **Gui-client user interface**
  - device instantiate and shutdown
  - single device command and configuration
  - interface to Project navigation
  - …
- **Projects**
  - scene and macro interface
  - allows group actions
  - …
- **Scenes**
  - customizable, cuts away all but needed
  - drag & drop dynamic creation
  - …

Gui-client primary tool used by experiments

European XFEL

Data Drivers: Infrastructure

Offline storage resources

- Fast Access:
  - Hardware: IBM Elastic Storage System
  - Software: GPFS
  - Current capacity: 14PB
- Large Capacity
  - Hardware: DELL Systems
  - Software: dCache
  - Current capacity: 16PB
- Archive:
  - Tape based

Data Drivers: Infrastructure

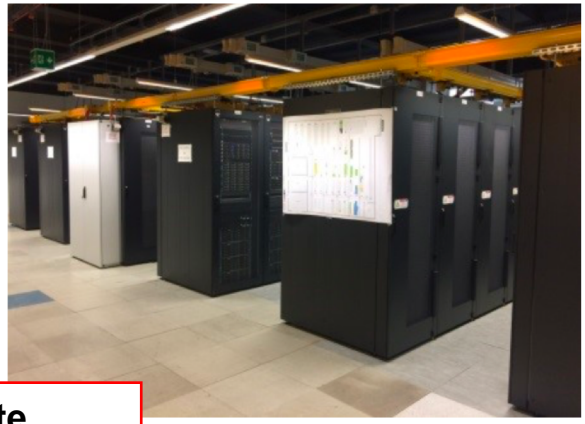
Offline computing Cluster

- Available resources:
  - R_{peak} = ~570TFlops,
  - 254 nodes
  - ~9200 cores
- Including recent upgrade (Oct 2019):
  - **R_{peak} = ~215Tflops**
  - **72 nodes**
  - **2592 cores**
- Next upgrade planned for spring 2020
  - 50 nodes

- Interactive login nodes
  - max-display nodes available from outside network
  - **4 additional login nodes in production**
- Slurm scheduler
  - Improved scheduling policies
  - **High priority for calibration jobs**
  - Reservation of computing resources for users of “current” beamtime
- Recommended access
  - FastX client
- Jhub service available

Strong collaboration with DESY computing center

Data Drivers: Infrastructure



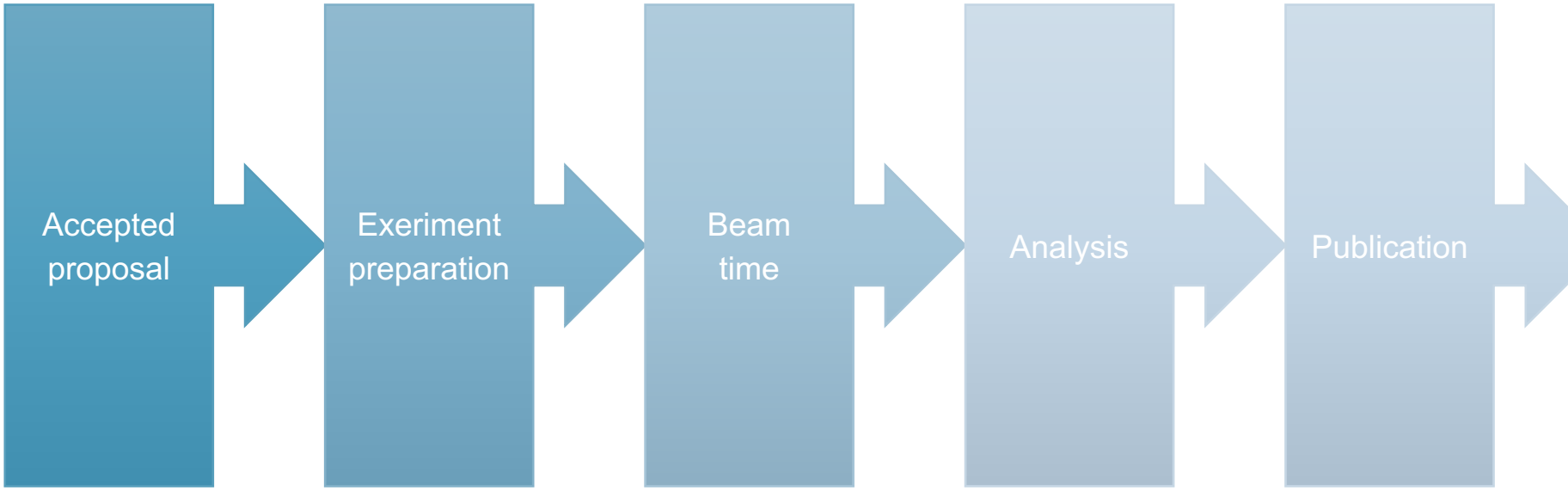
Offline computing Cluster

- Available resources:
  - R_{peak} = ~570TFlops,
  - 254 nodes
  - ~9200 cores
- Including recent upgrade (Oct 2019):
  - **R_{peak} = ~215Tflop**
  - **72 nodes**
  - **2592 cores**
- Next upgrade planned for spring 2020
  - 50 nodes

- Interactive login nodes
  - max-display nodes available from outside network
  - **4 additional login nodes in production**
- Slurm scheduler
  - Improved scheduling policies
  - **High priority for calibration jobs**
  - Reservation of computing resources for users of “current” beamtime
- Recommended access
  - FastX client
- Jhub service available

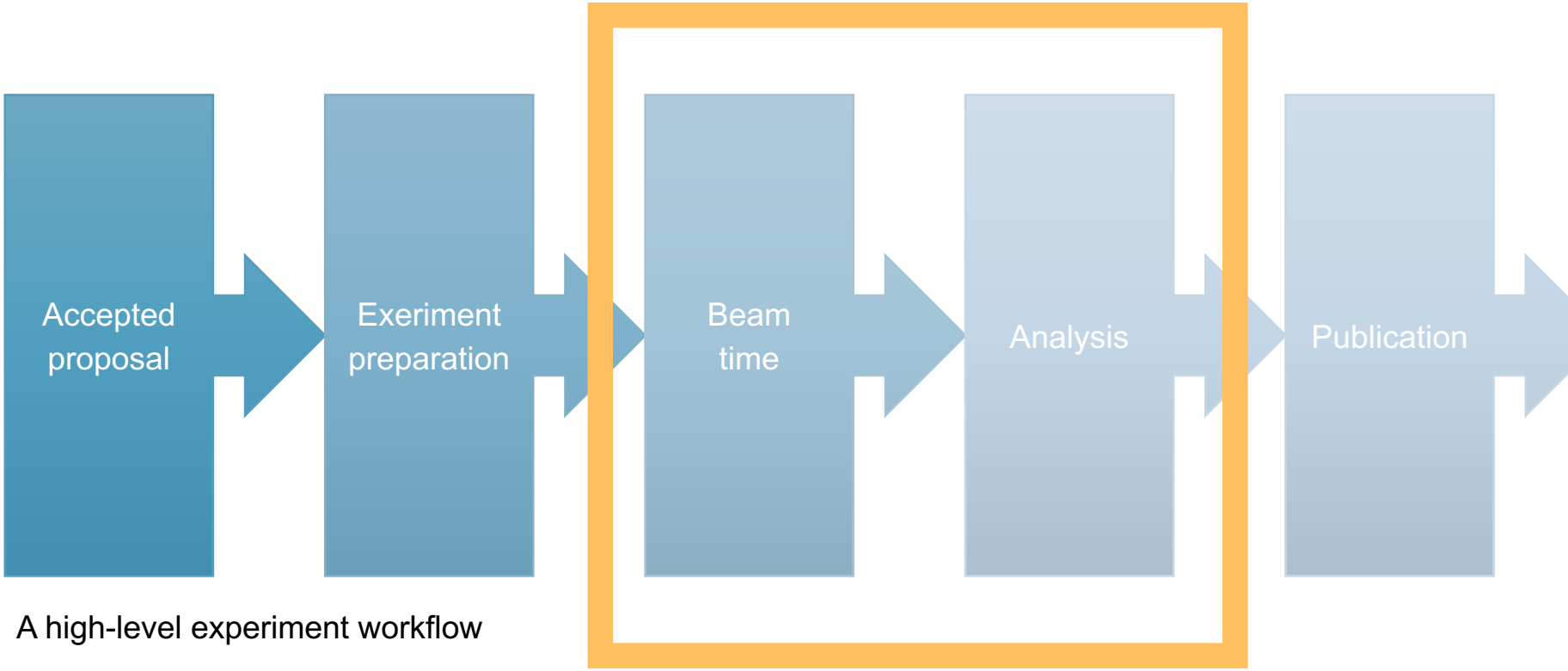
Most users have to analyze non-reduced data on-site

Data Solutions



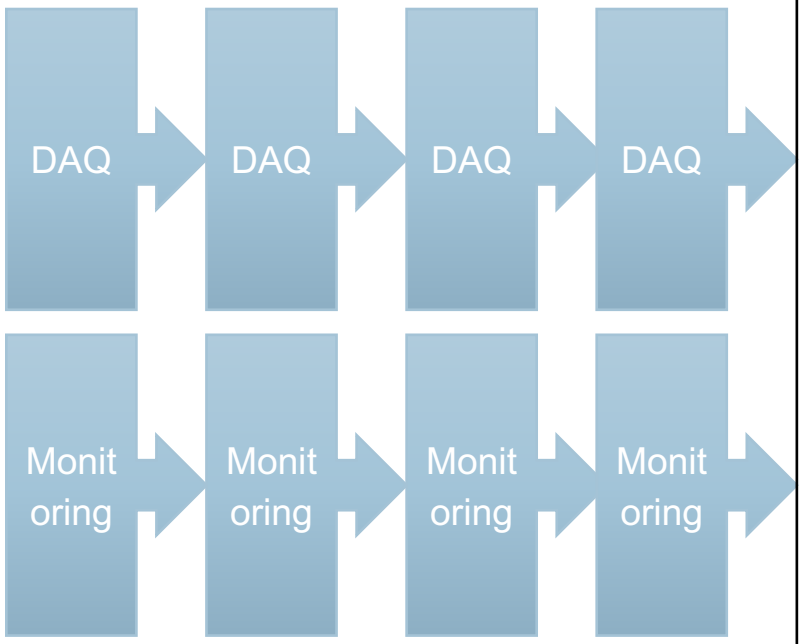
A high-level experiment workflow

Data Solutions



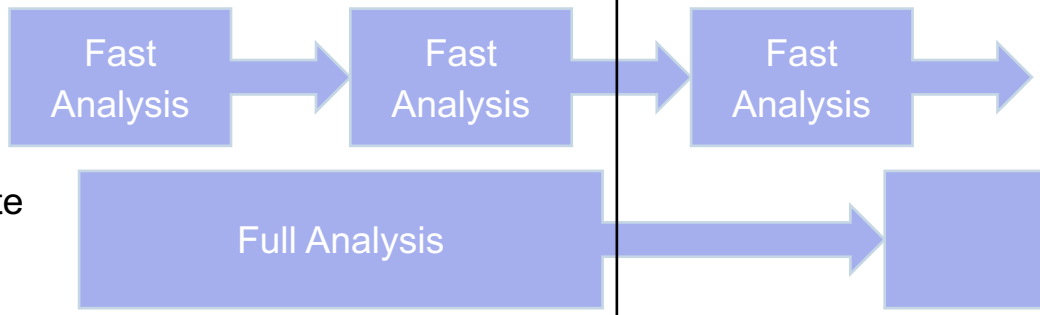
A high-level experiment workflow

Data Solutions



Post beamtime

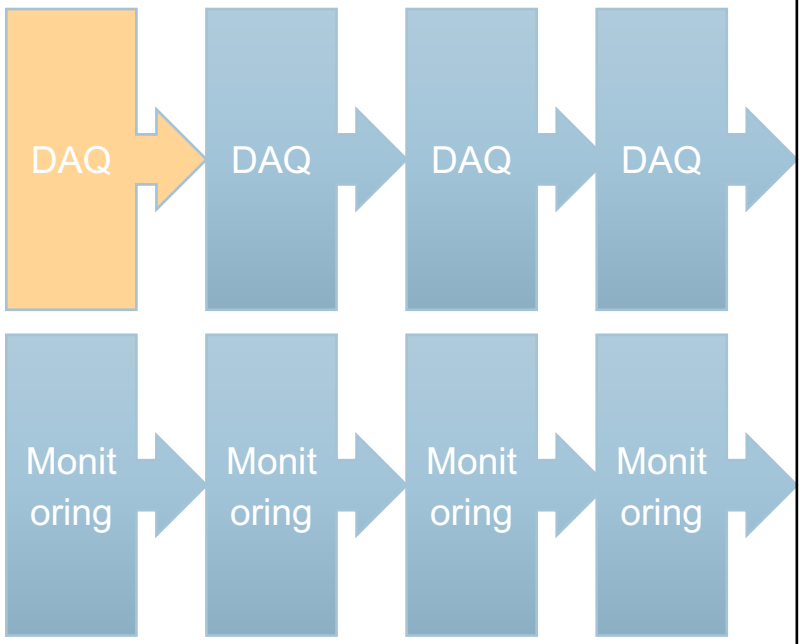
- Users have remote access
- **FAIR Reproducability** is important:
  - Metadata
  - Calibration parameters
  - Tagging of all software involved
  - Containerization + Virtualization
  - Docker
  - Conda
  - ...



Beamtime

- Users on site

Data Solutions: Data Acquisition

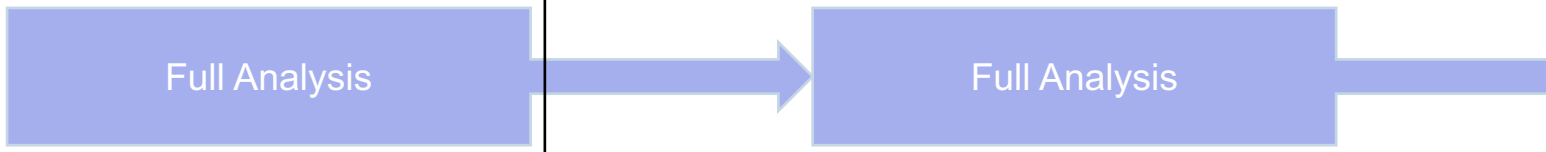


Post beamtime

- Users have remote access
- **FAIR Reproducability** is important:
  - Metadata
  - Calibration parameters
  - Tagging of all software involved
  - Containerization + Virtualization
  - Docker
  - Conda
  - ...

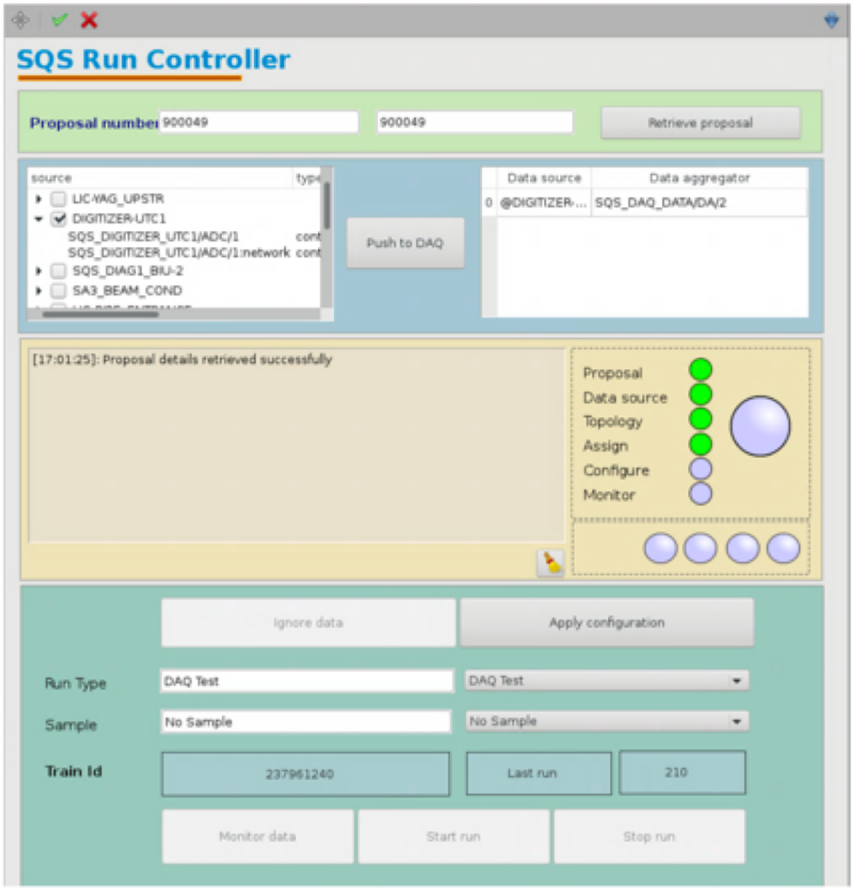
Beamtime

- Users on site



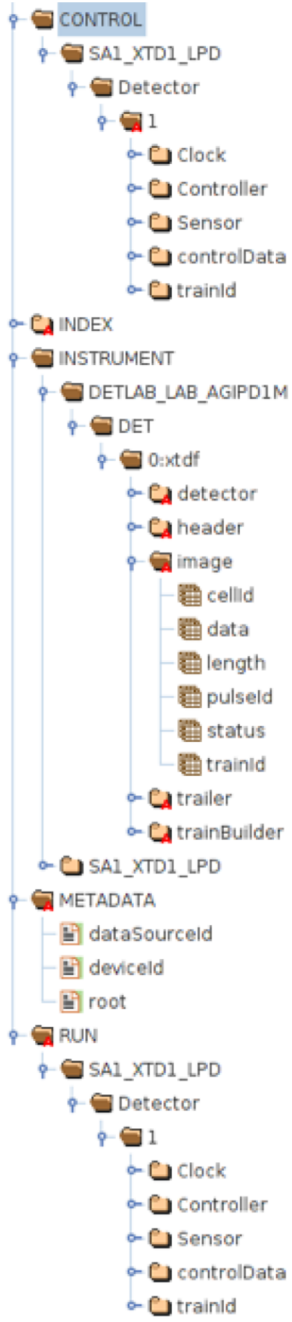
Data Solutions: Data Acquisition

- Distinguish between **user** data acquisition and **facility** “housekeeping“ data
  - User data, needs to conform to *Scientifc Data Policy*
    - ► Long term data curation
    - ► Open access
    - ► Embargo period with restricted access
    - ► Data accessible by European XFEL staff during embargo period
    - ► All users need to agree to the policy as part of beamtime application
  - Facility data, only used for facility purposes, not tied to individual experiments



Data Solutions: Data Acquisition

- User data is stored in HDF5 files
  - Structure reflects control and data source topology of the facility
- Data can be train or pulse resolved
  - Associated meta data included for timing correlation
- Train frequency is assumed
  - If for a given train no slow data is updated a copy is created
  - Assures that all table rows in a file line up in time
- File organized into experiment „runs“
  - A run contains many „sequence“ files
    - ► Files of managable size, aim ~20GB for MHz detectors
    - ► Smaller if only slow data or 10Hz cameras

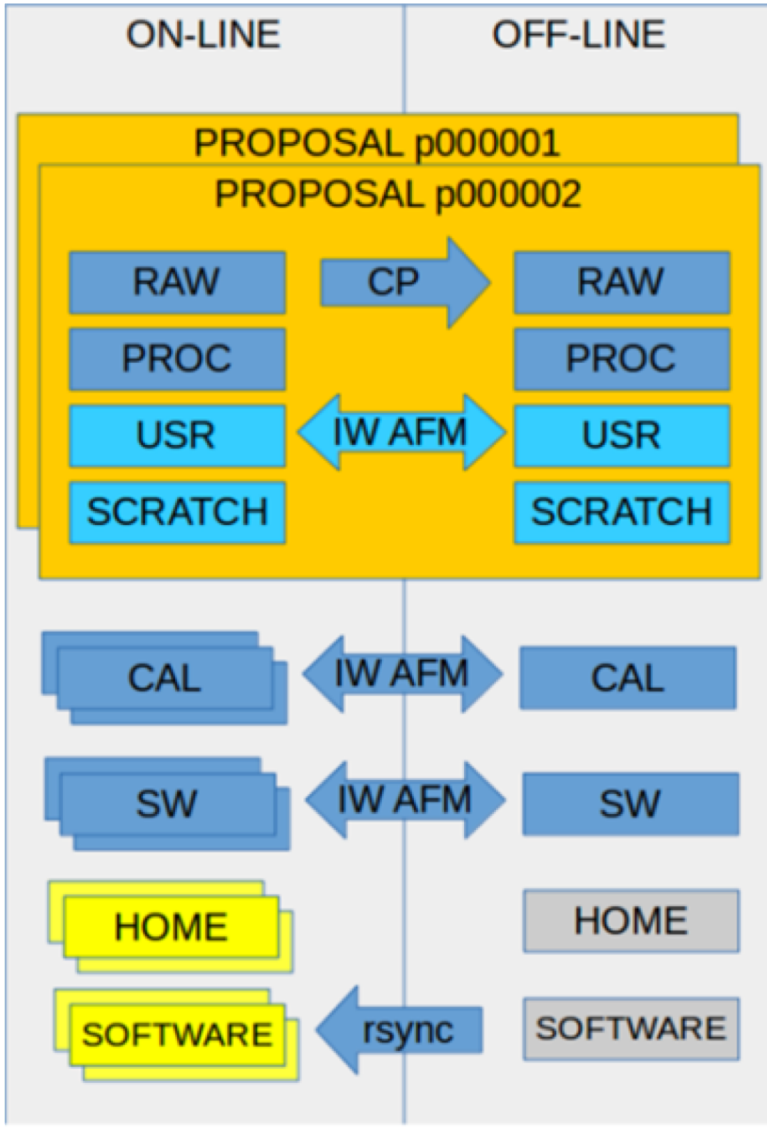


See: https://in.xfel.eu/readthedocs/docs/data-analysis-user-documentation/en/latest/index.html

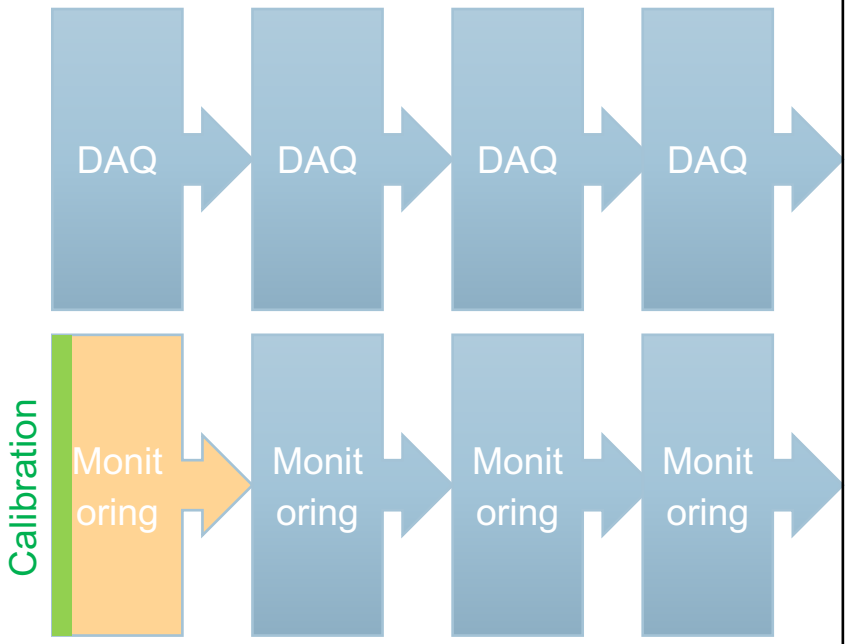
Data Solutions: Data Acquisition

Data placement

- Proposal specific (on request)
  - RAW – raw data from detector
  - PROC – processed data
  - USR – user data, scripts, programs
  - SCRATCH – intermediate results
- Proposal independent
  - CAL – calibration constants
  - SW – internal control software
  - HOME – $home folder of computing clusters
  - SOFTWARE – analysis software, python, matlab, …
  - ….



Data Solutions: Calibration

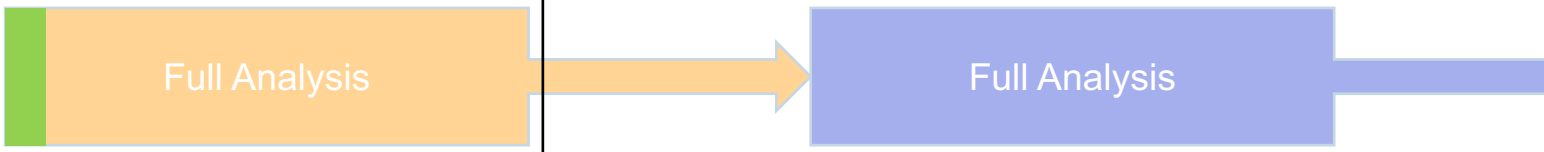


Post beamtime

- Users have remote access
- **FAIR Reproducability** is important:
  - Metadata
  - Calibration parameters
  - Tagging of all software involved
  - Containerization + Virtualization
  - Docker
  - Conda
  - ...

Beamtime

- Users on site

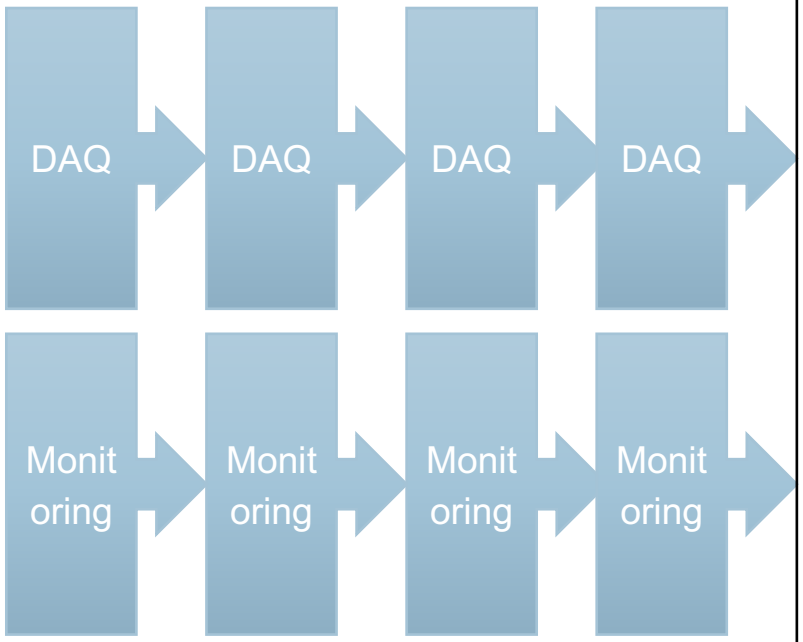


Data Solutions: Calibration and Correction (2D Detectors)

- ***Retrieve parameters for correction:*** resolve detector conditions, point in time leading to most appropriate *parameters*
- ***Corrected and calibrated data is main data product:*** needs to be available as soon as possible and reliable
- ***Produce correction parameters:*** characterize detector and select necessary subset of information. Perform quality assessment
- ***Manage correction parameters:*** centrally persist, categorize, select parameters
- **Optimize corrections according to specific scientific needs** e.g. real vs integer photon numbers, split event corrections
- ***Agility:*** support short cycles from prototyping, testing to production deployment where ever possible

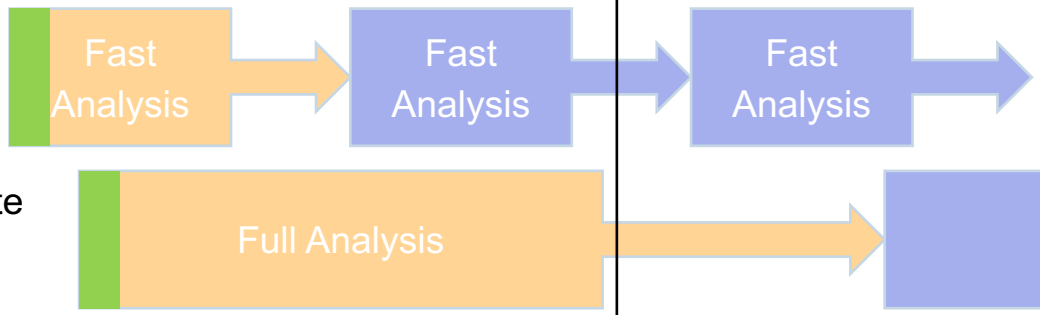
Python: fast development cycles, good data analysis capabilities

Data Solutions



Post beamtime

- Users have remote access
- **FAIR Reproducability** is important:
  - Metadata
  - Calibration parameters
  - Tagging of all software involved
  - Containerization + Virtualization
  - Docker
  - Conda
  - ...



Beamtime

- Users on site

Data Solutions: Offline Calibration Environment

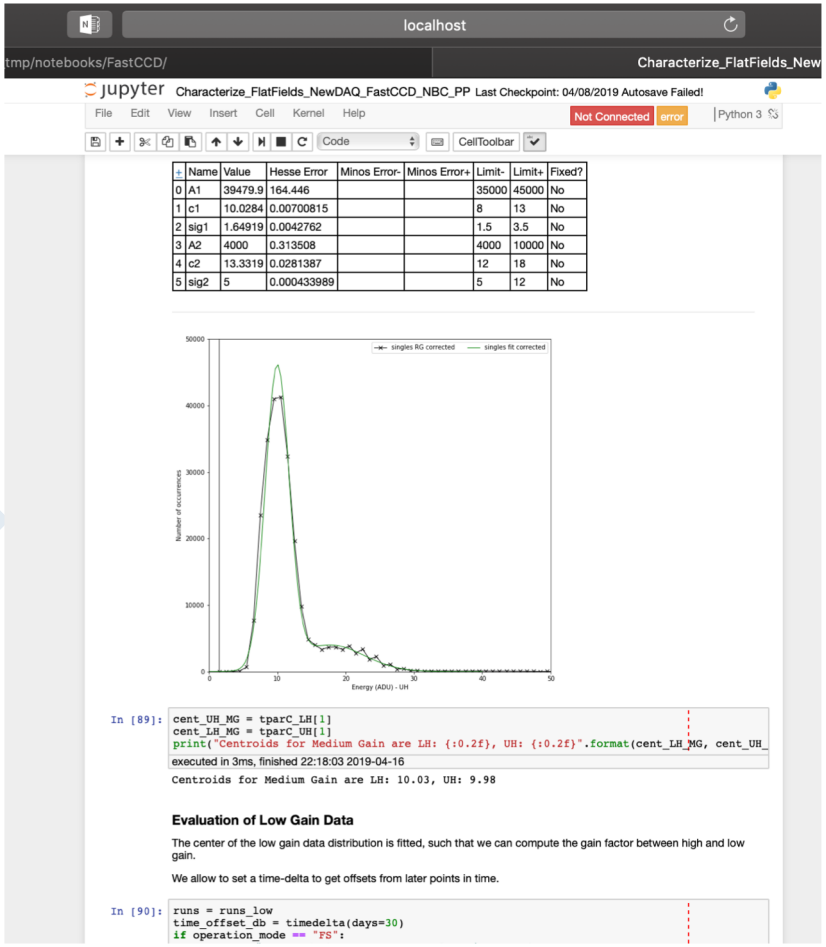
jupyter

Python

NumPy

matplotlib

SciPy

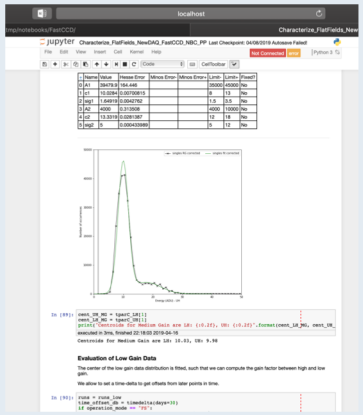


	Name	Value	Hesse Error	Minos Error-	Minos Error+	Limit-	Limit+	Fixed?
0	A1	39479.9	164.446			35000	45000	No
1	c1	10.0284	0.00700815			8	13	No
2	sig1	1.64919	0.0042762			1.5	3.5	No
3	A2	4000	0.313508			4000	10000	No
4	c2	13.3319	0.0281387			12	18	No
5	sig2	5	0.000433989			5	12	No

Rapid, interactive development in Jupyter Notebook

Data Solutions: Offline Calibration Environment

jupyter
Python
NumPy
matplotlib
SciPy



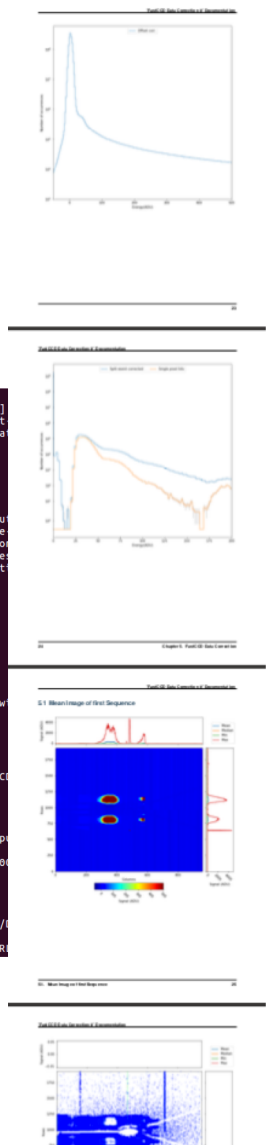
SPHINX
PYTHON DOCUMENTATION GENERATOR

nbparameterise

slurm
workload manager

```
[xcal@max-exfl016 ~]$ xfel-calibrate FASTCCD CORRECT --help
usage: xfel-calibrate [-h] [--no-cluster-job] [--report-to str]
                      [--priority int] [--in-folder str] [--out
                      [--path-template str] [--run int] [--h5pa
                      [--h5path-t str] [--h5path-cntrl str]
                      [--cluster-profile str] [--cpuCores int]
                      [--operation-mode str]
                      [--split-evt-primary-threshold float]
                      [--split-evt-secondary-threshold float]
                      [--split-evt-mip-threshold float]
                      [--cal-db-interface str] [--cal-db-timeou
                      [--sequences str [str ...]] [--chunk-size
                      [--overwrite] [--do-pattern-classificatio
                      [--sequences-per-node int] [--limit-image
                      [--correct-offset-drift] [--use-dir-creat
                      [--time-offset-days int]
                      [--photon-energy-gain-map float]
                      [--fix-temperature float]
                      [--flipped-between str [str ...]]
                      DETECTOR TYPE

# FastCCD Data Correction ##

Authors: I. Klaćková, S. Hauf, Version 1.0,

The following notebook provides correction of images acquired w
FastCCD.

positional arguments:
  DETECTOR              The detector to calibrate
  TYPE                  Type of calibration:
                        AGIPD,LPD,PNCCD,GENERIC,TUTORIAL,FASTCC

optional arguments:
  -h, --help            show this help message and exit
  --no-cluster-job      Do not run as a cluster job
  --report-to str       Filename (and optionally path) for outp
  --priority int
  --in-folder str       Default: /gpfs/exfel/exp/SCS/201930/p90
  --out-folder str      output folder. Default:
                        /gpfs/exfel/data/scratch/xcal/test/
  --path-template str   Default: RAW-R{:04d}-DA05-S{{:05d}}.h5
  --run int             Default: 406
  --h5path str          Default: /INSTRUMENT/SCS_CDIDET_FCCD2M/
                        put/data/image
  --h5path-t str        Default: /CONTROL/SCS_CDIDET_FCCD2M/CTR
```

PDF reports



Data Solutions: Available Corrections

* Corrections applied by default

(**into add. Output)

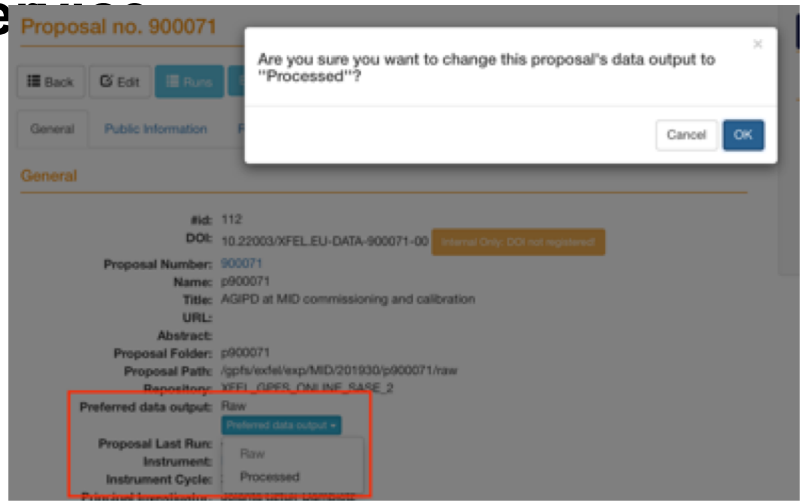
Parameter	AGIPD	DSSC	LPD	CCDs	ePIX	Jungfrau
Dark signal (cell nr,x,y)	X*	X*	X*	X*	X*	X*
Dark baseline shifts (x,y)	X			X (FastCCD)		
Noise N(cell nr, x,y)	X*	X*	X*	X*	X*	X*
Conversion gain (cell nr, x,y)	X		X	X*	(X)	X*
Gain transition region (cell nr, x,y)	X (phenom.)		X			
„Snowy“ pixels (x,y)	X					
Bad pixel map (cell nr,x,y)	X	X	X	X	X	X
Common mode				X**	X**	
Memory cell droop (cell nr, x,y)						
Charge splitting (x,y)	Requested			X**	X**	X
Charge transfer inefficiency CTI(x,y)				(X)		
Quantum efficiency (x,y)						
Alignment	X (CFEL dataset)	X	X up to Quadrant	X		

Data Solutions: Available Standard Characterizations

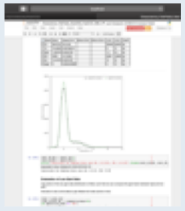
* Can be triggered by Instruments

Parameter	AGIPD	DSSC	LPD	CCDs	ePIX	Jungfrau
Dark signal (cell nr,x,y)	X*	X*	X*	X*	X*	X*
Noise N(cell nr, x,y)	X*	X*	X*	X*	X*	X*
Conversion gain (cell nr, x,y)						
- using charge injection	X		X			
- using X-rays	X		X	X		X
Bad pixel map (cell nr,x,y)	X*	X*	X*	X*	X*	X*

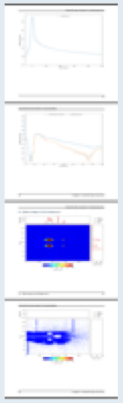
Data Solutions: Calibration Web Se



jupyter
Python
NumPy
matplotlib
SciPy



SPHINX
slurm workload manager
nbparameterise



Users trigger (re) calibration via MDC

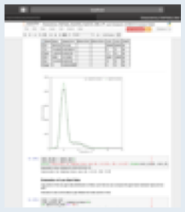
Proposal Runs

Run Number (alias)	Run type	Sample Name	Start date	Run status	Data Assessment	Calibration	Run Comment	Edit
0448	Calibration - Dark LG	No Sample	2019-03-19 17:30:27 +0100	Closed	Run Quality			
0447	Calibration - Dark MG	No Sample	2019-03-19 17:29:03 +0100	Closed	Run Quality			
0446	Calibration - Dark HG	No Sample	2019-03-19 17:27:40 +0100	Closed	Run Quality			
0445	Calibration - Dark LG	No Sample	2019-03-18 04:43:44 +0100	Closed	Good			
0444	Calibration - Dark MG	No Sample	2019-03-18 04:42:21 +0100	Closed	Good	(Re)calibrate		
0443	Calibration - Dark HG	No Sample	2019-03-18 04:40:57 +0100	Closed	Good			
0442	Calibration - Dark HG	No Sample	2019-03-18 04:39:27 +0100	Closed	Not interesting			
0441	X-ray Diffraction	Silica	2019-03-18 04:21:27 +0100	Closed	Good			

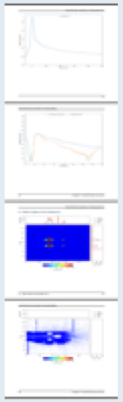
1
2
3

Data Solutions: Calibration Web Service

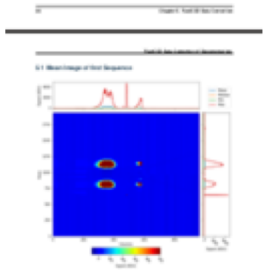
jupyter
Python
NumPy
matplotlib
SciPy

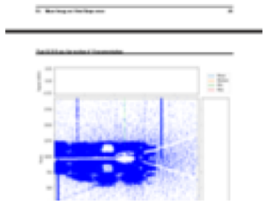


SPHINX
slurm workload manager
nbparameterise

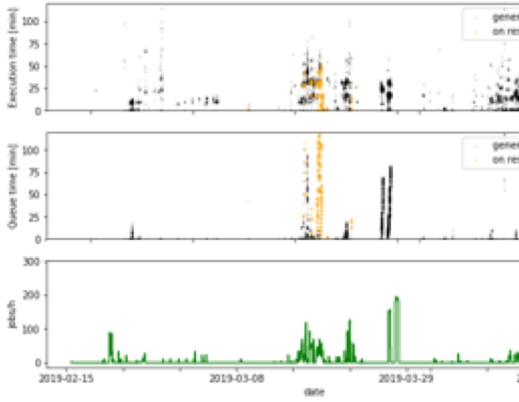


PDF reports





Performance optimization

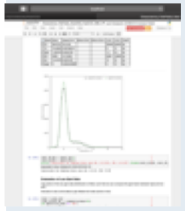


Data Solutions: Calibration Web Service – Workflow

Established tools

jupyter
Python
NumPy
matplotlib
SciPy

Development

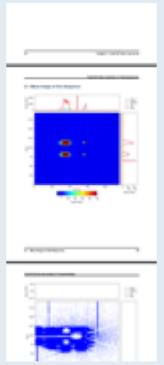


Expert Usage

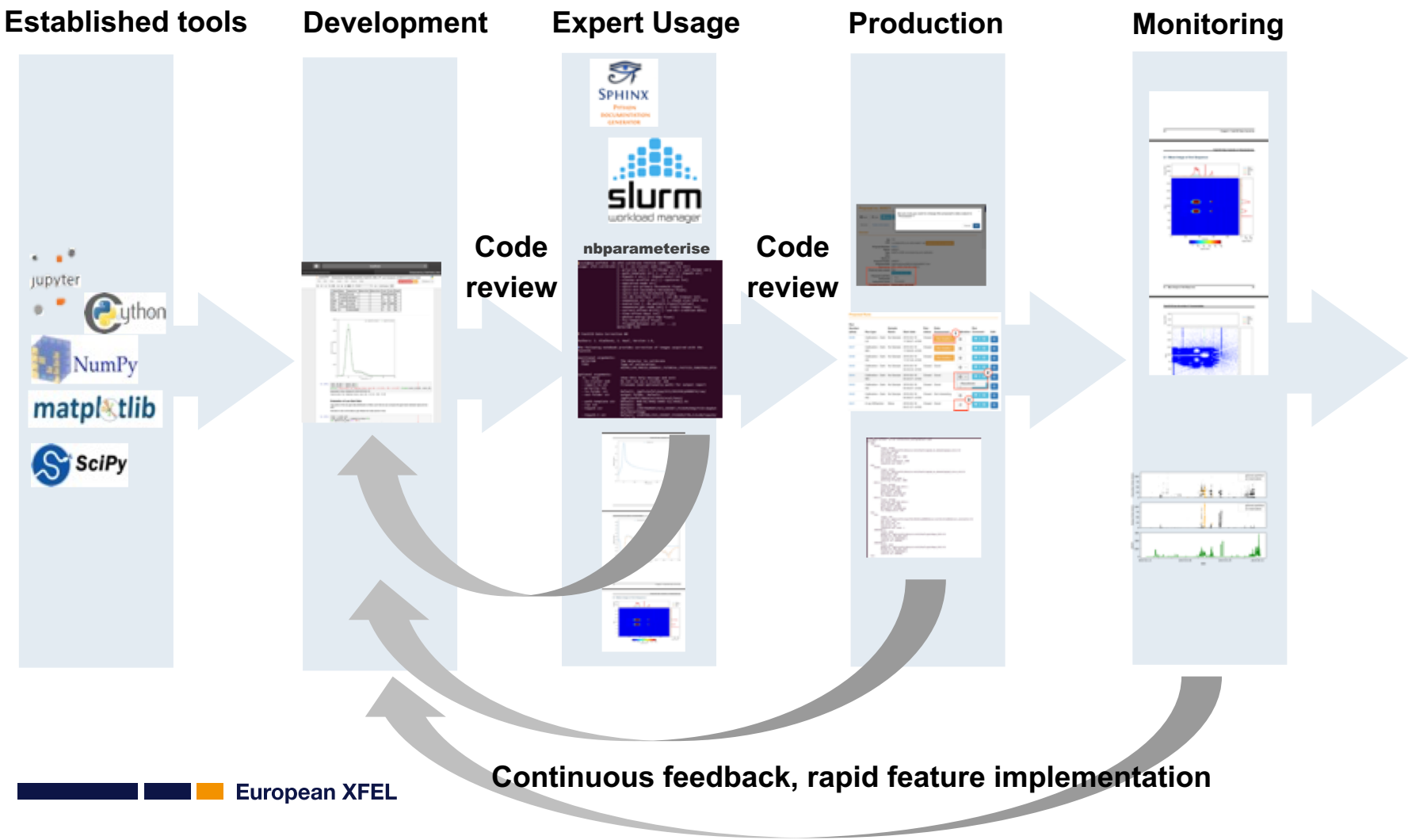
SPHINX
slurm workload manager
nbparameterise

Production

Monitoring & QA

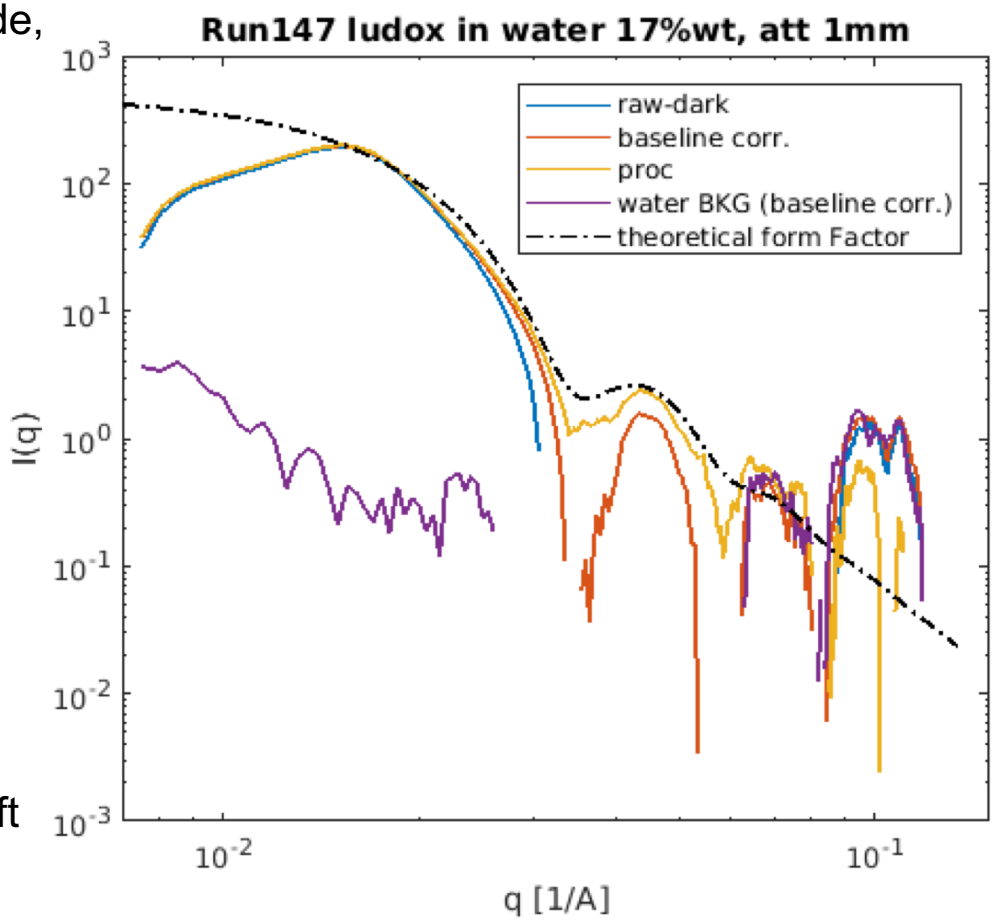


Data Solutions: Calibration Web Service – Workflow



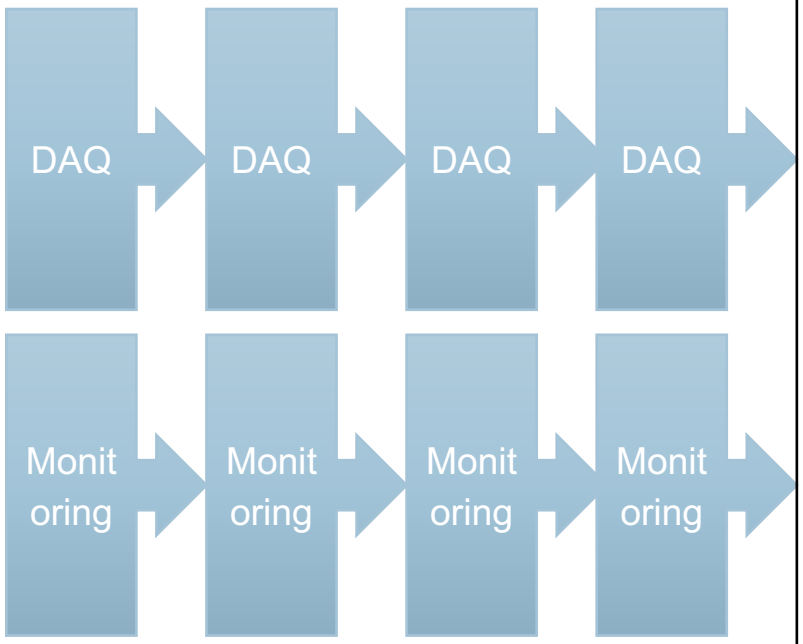
Data Solutions: Example of Beamline/User Interaction

- Corrections implemented on user side, to compare against facility corrected data
  - raw-dark
    - ►offset subtraction
  - baseline corr.
    - ►Correct offset drift via border region median
- proc:
  - Third iteration of facility provided corrections
  - Offset correction and baseline drift compensation



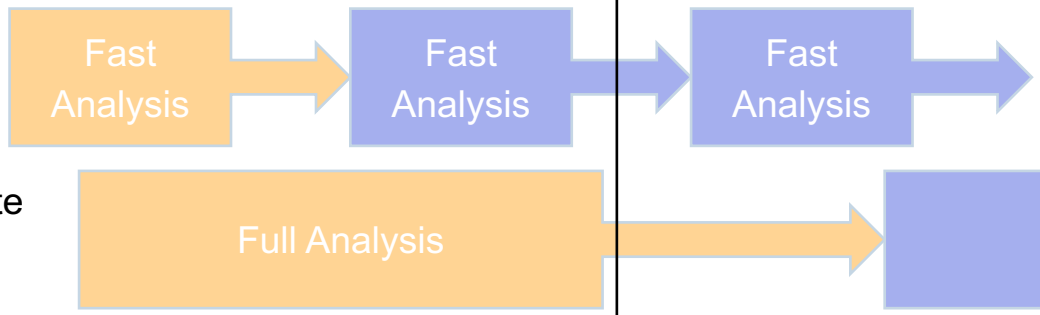
Courtesy: F. Lehmkühler et al.

Data Solutions: Data Access and Correlation



Post beamtime

- Users have remote access
- **Reproducability** is important:
  - Metadata
  - Calibration parameters
  - Tagging of all software involved
  - Containerization + Virtualization
  - Docker
  - Conda
  - ...



Beamtime

- Users on site

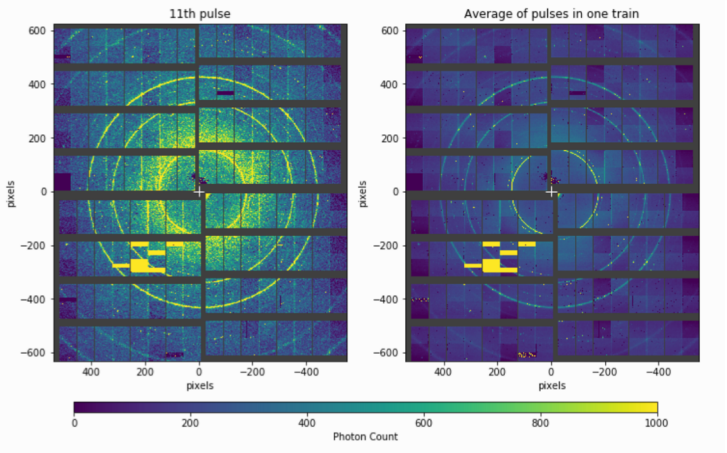
Data Solutions: Simplifying Data Access and Correlation

- extra-data:
  - Python library for iterating over XFEL run data on a train-wise basis
  - Select data sources by name, filter by index, trainId, ...
  - Abstracts away indexing needed within files for time correlation
  - https://extra-data.readthedocs.io/en/latest/
- extra-geom
  - Geometry handling of segmented detectors
- extra-data-validate
  - Validy check for data contents

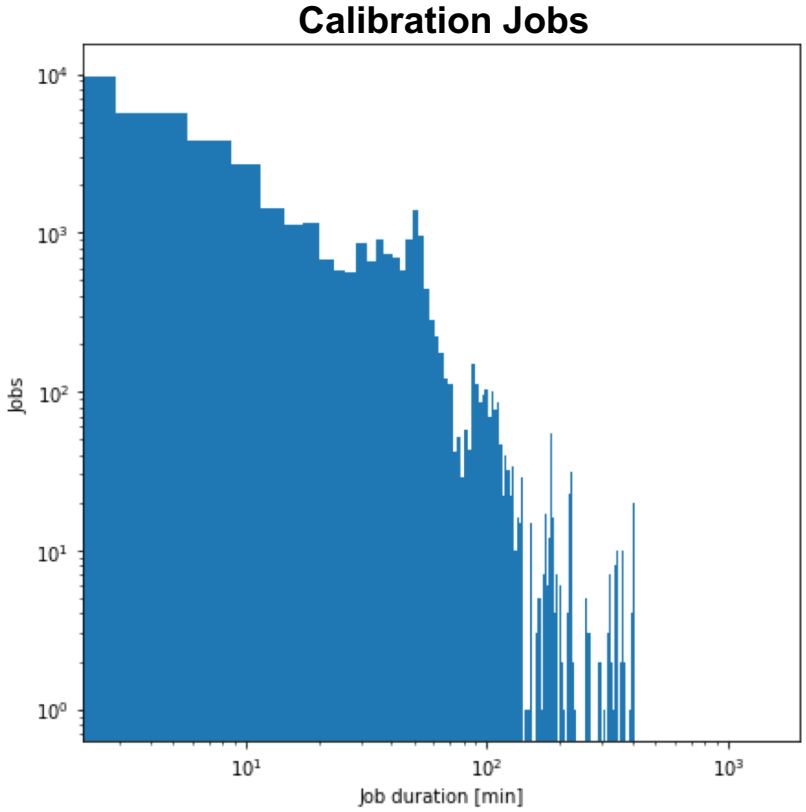
```python
from extra_data import open_run, RunDirectory, H5File

# Find a run on the Maxwell cluster
run = open_run(proposal=700000, run=1)

# Open a run with a directory path
run = RunDirectory("/gpfs/exfel/exp/XMPL/201750/p700000/raw/r0001")

# Open an individual file
file = H5File("RAW-R0017-DA01-S00000.h5")
```

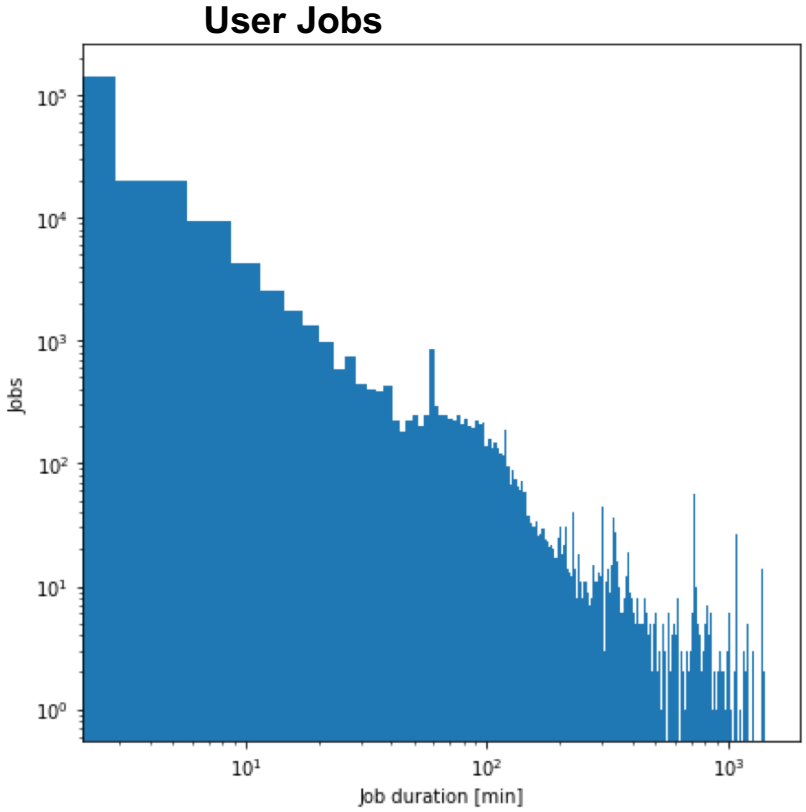
```python
for train_id, data in run.select("*/DET/*", "image.data").trains():
    mod0 = data["FXE_DET_LPD1M-1/DET/0CH0:xtdf"]["image.data"]
```

```python
geom.plot_data_fast(stacked_pulse, vmin=0, vmax=1000)
```



Data Solutions: Maxwell Usage

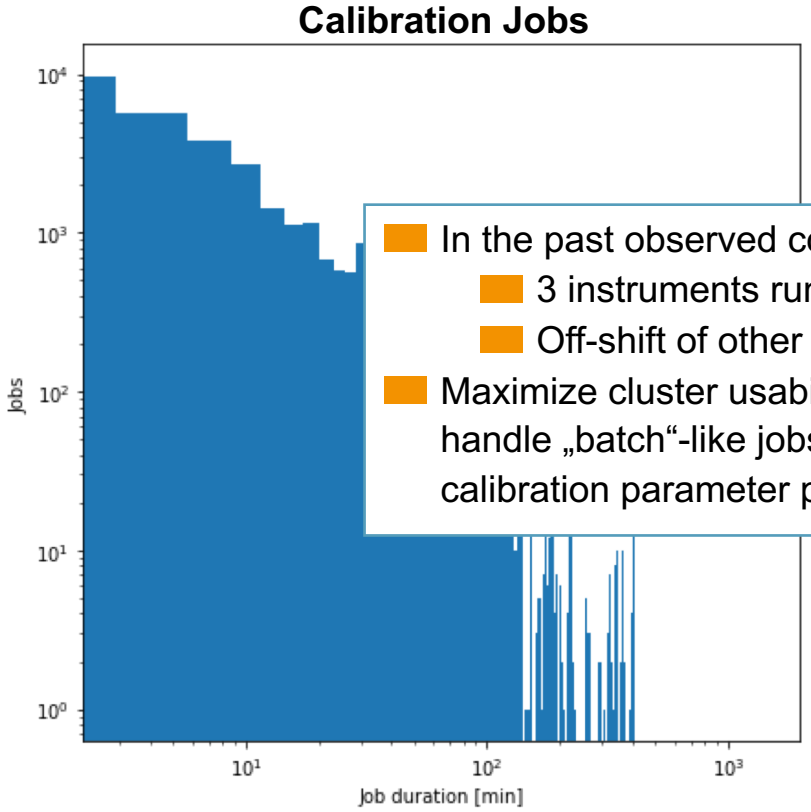


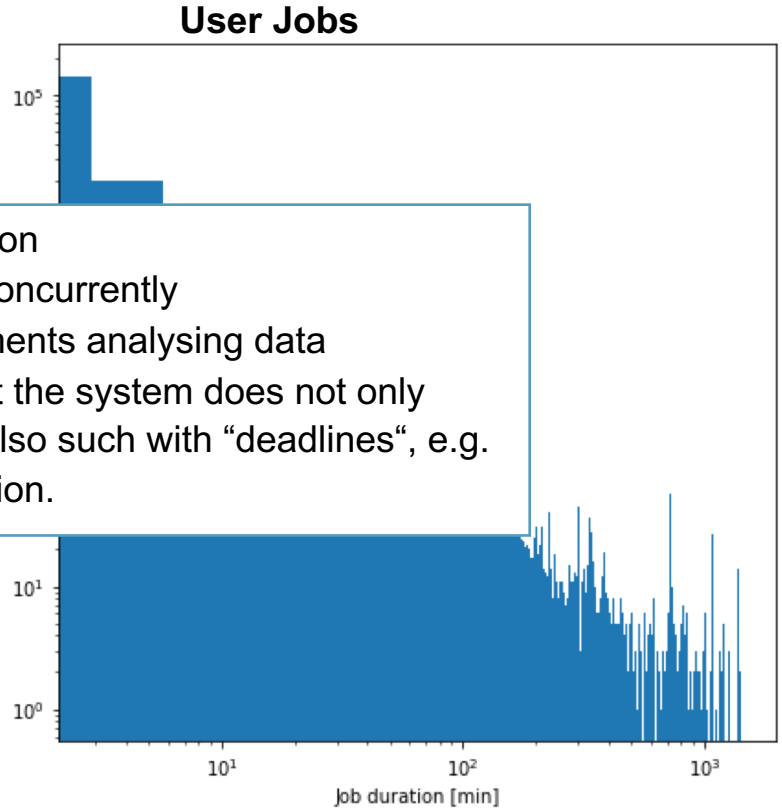


- Total number of calibration jobs: 38553
- Total job run time: 17908.3 hours
- 0.45 hours/job

- Total number of user jobs: 194940
- Total job run time: 61528.0 hours
- 0.30 hours/job

Data Solutions: Maxwell Usage



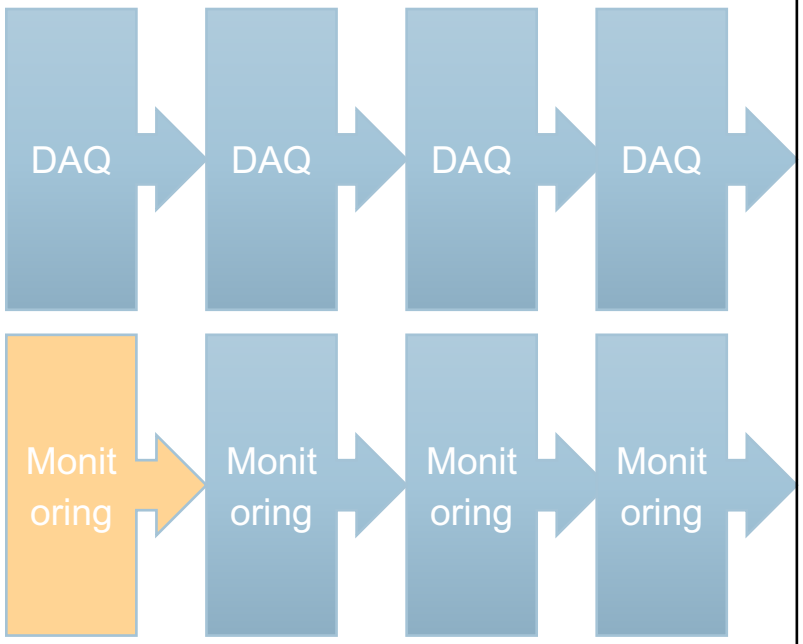


- In the past observed congestion
  - 3 instruments running concurrently
  - Off-shift of other instruments analysing data
- Maximize cluster usability, but the system does not only handle „batch“-like jobs, but also such with “deadlines“, e.g. calibration parameter production.

- Total number of calibration jobs: 38553
- Total job run time: 17908.3 hours
- 0.45 hours/job

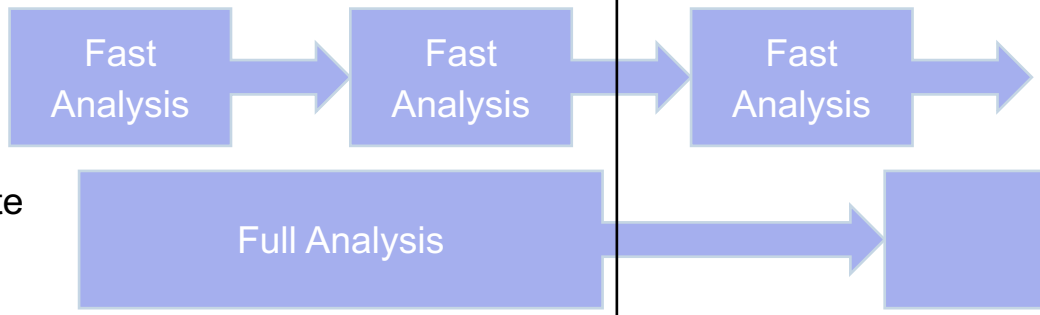
- Total number of user jobs: 194940
- Total job run time: 61528.0 hours
- 0.30 hours/job

Data Solutions: Online Monitoring



Post beamtime

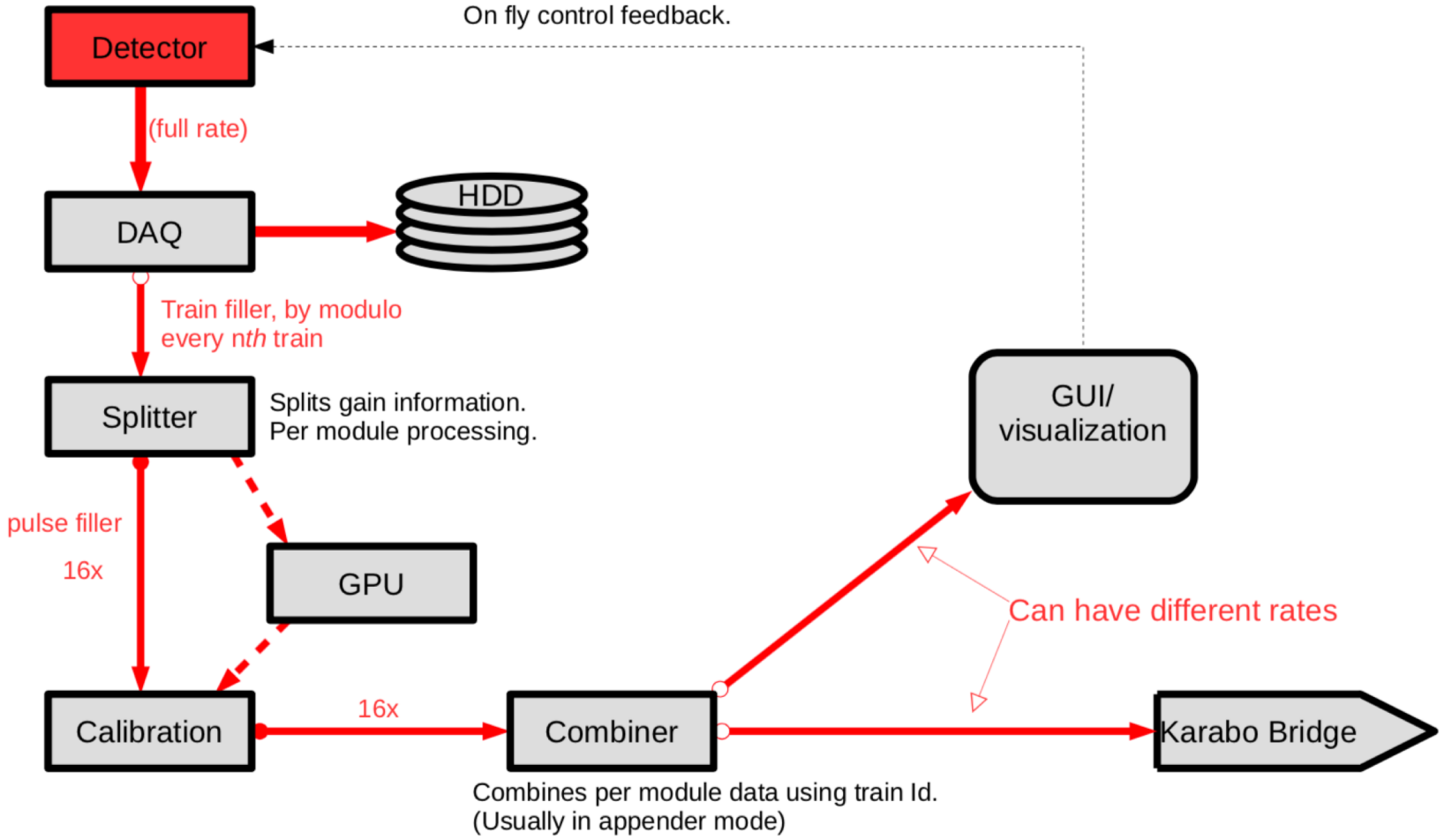
- Users have remote access
- **Reproducability** is important:
  - Metadata
  - Calibration parameters
  - Tagging of all software involved
  - Containerization + Virtualization
  - Docker
  - Conda
  - ...



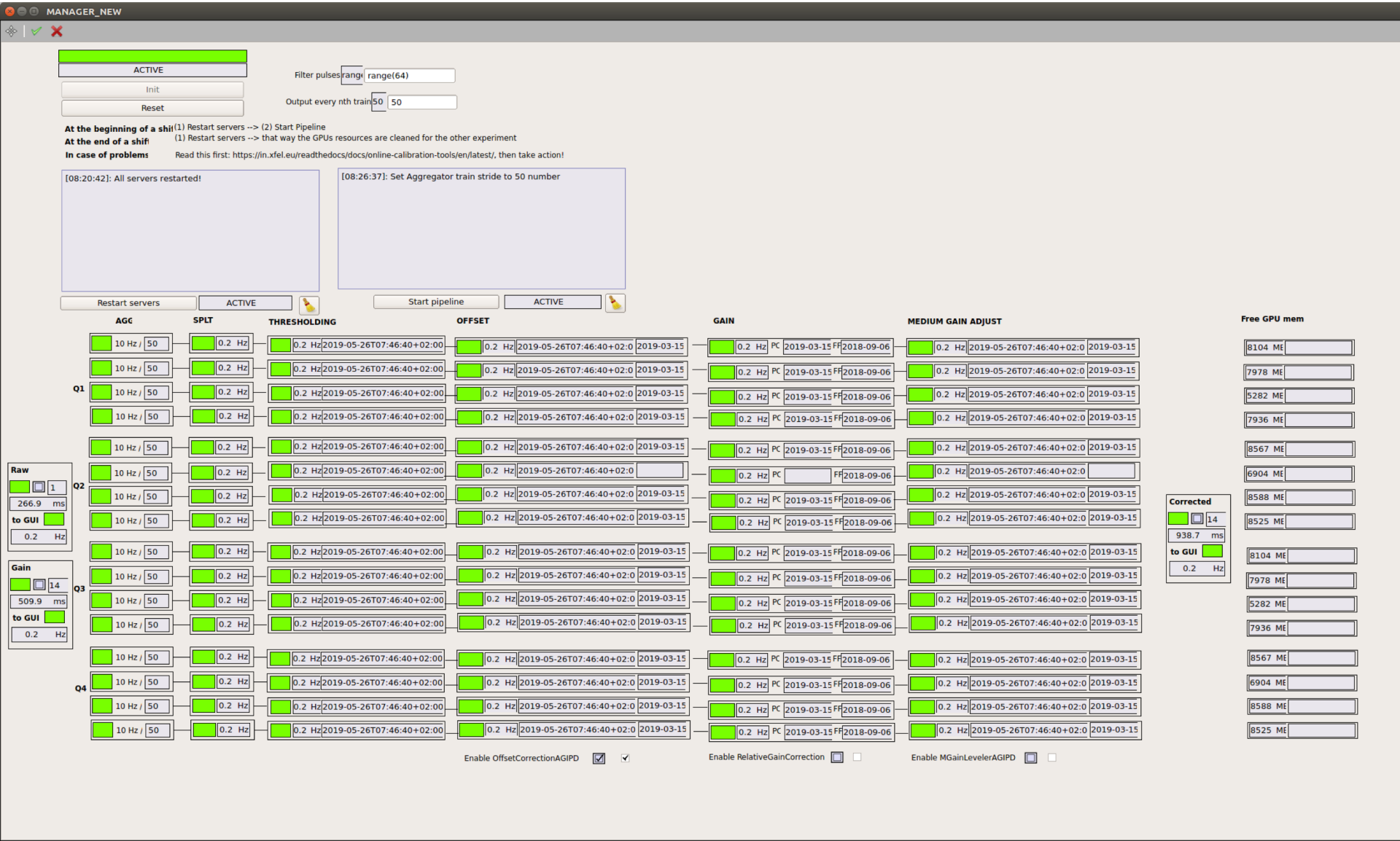
Beamtime

- Users on site

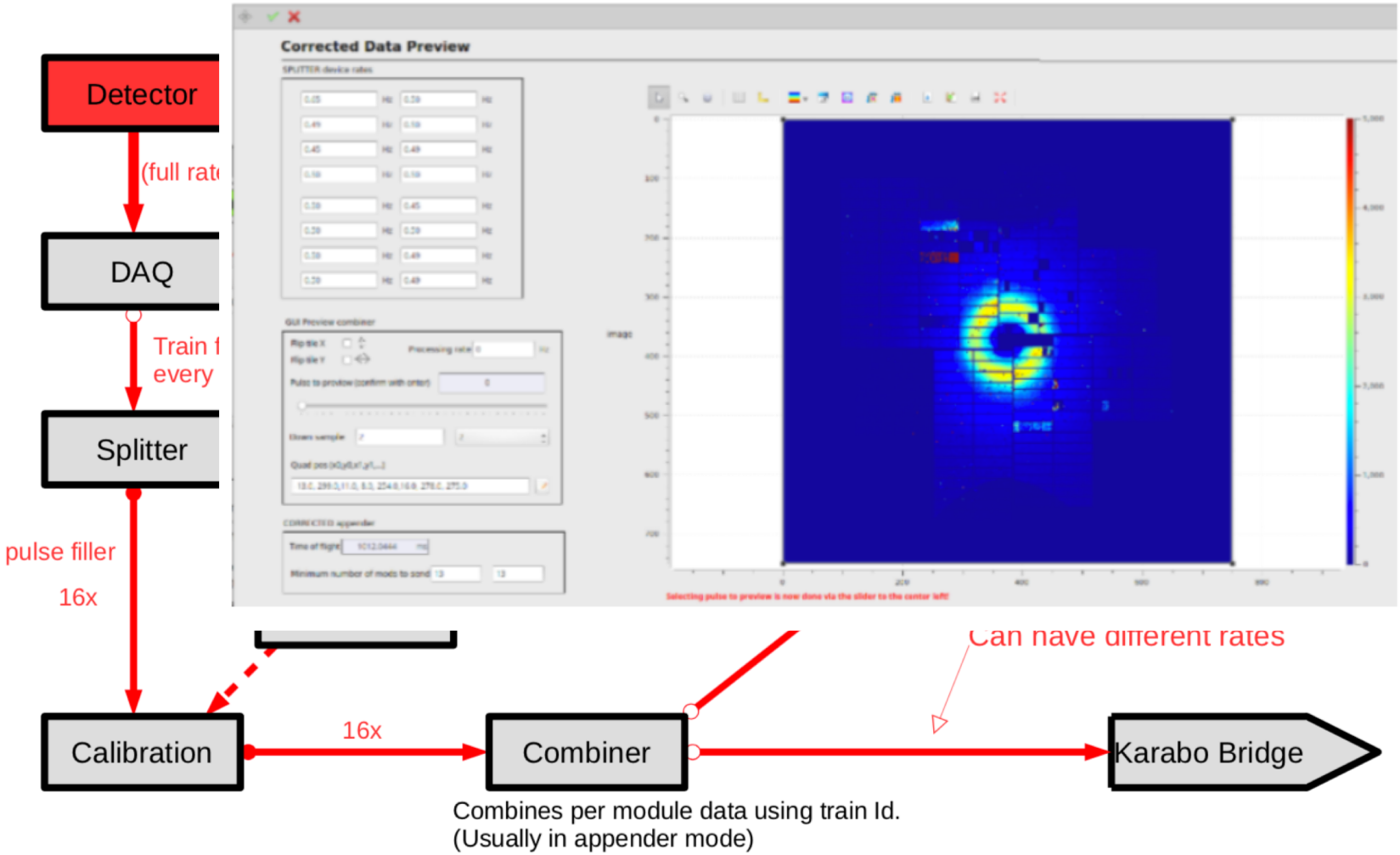
Data Solutions: Online Views – Controlling the Experiment



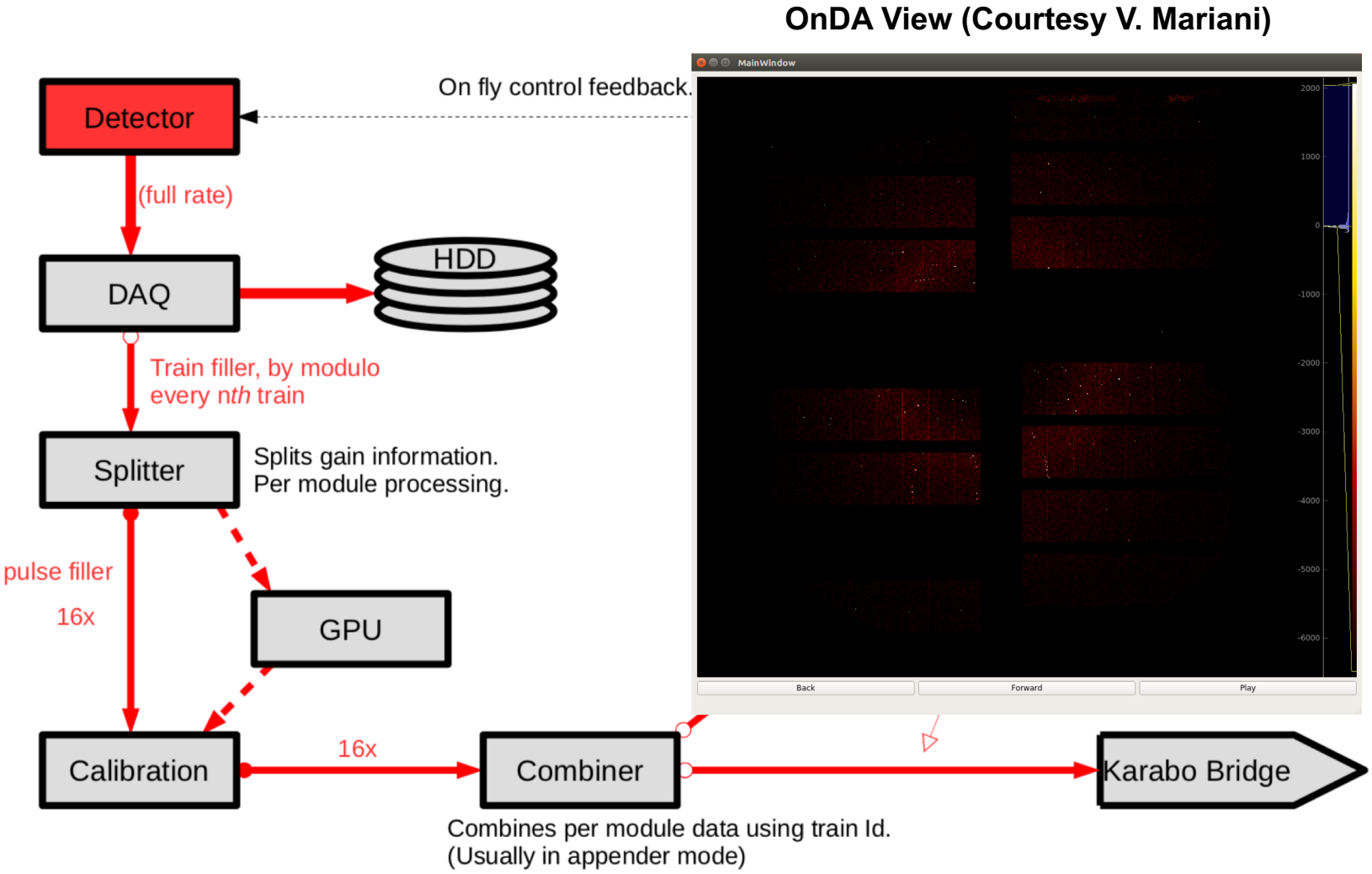
Data Solutions: Online Views – Pluggable Pipelines



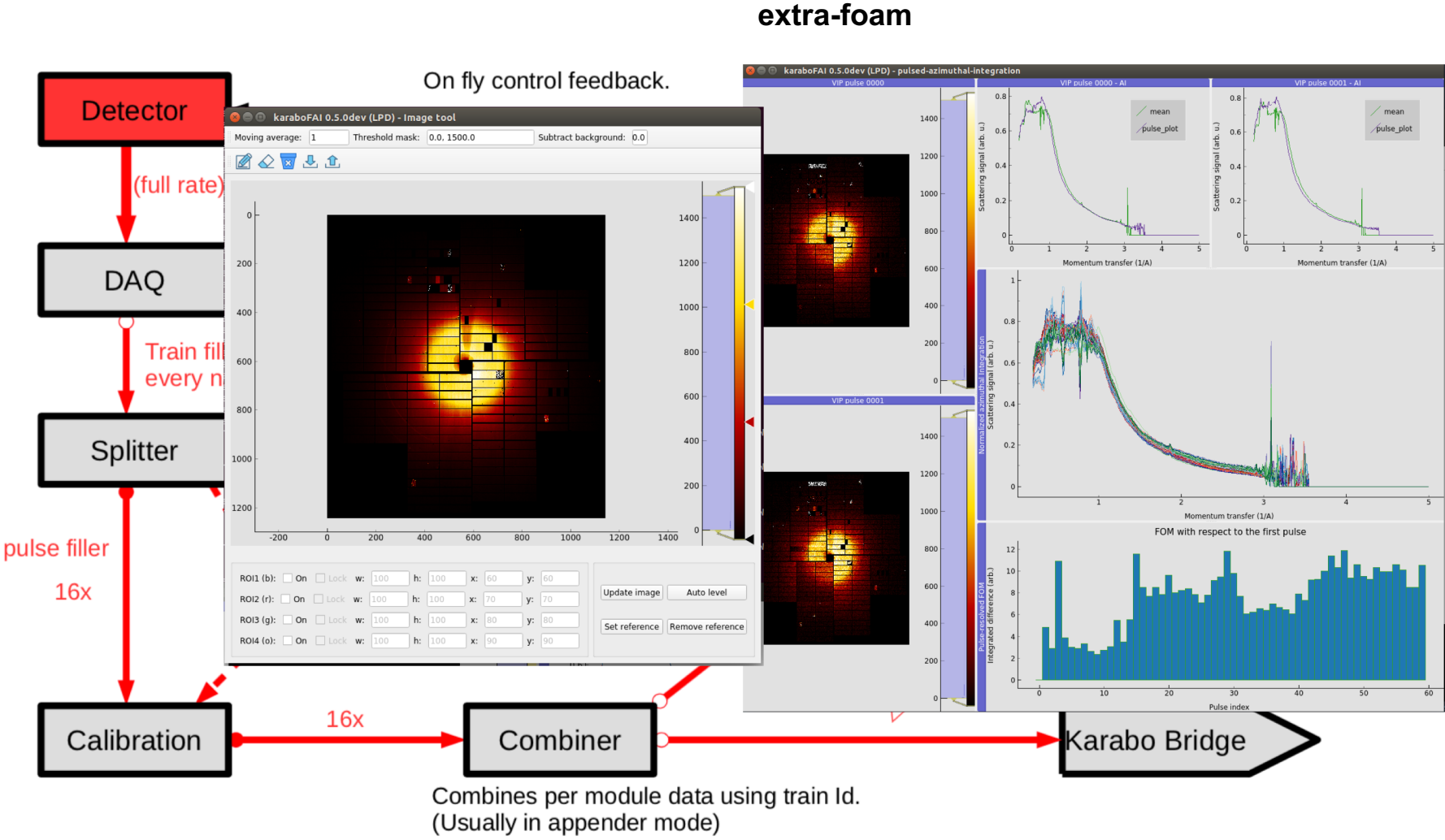
Data Solutions: Online Views – Previews



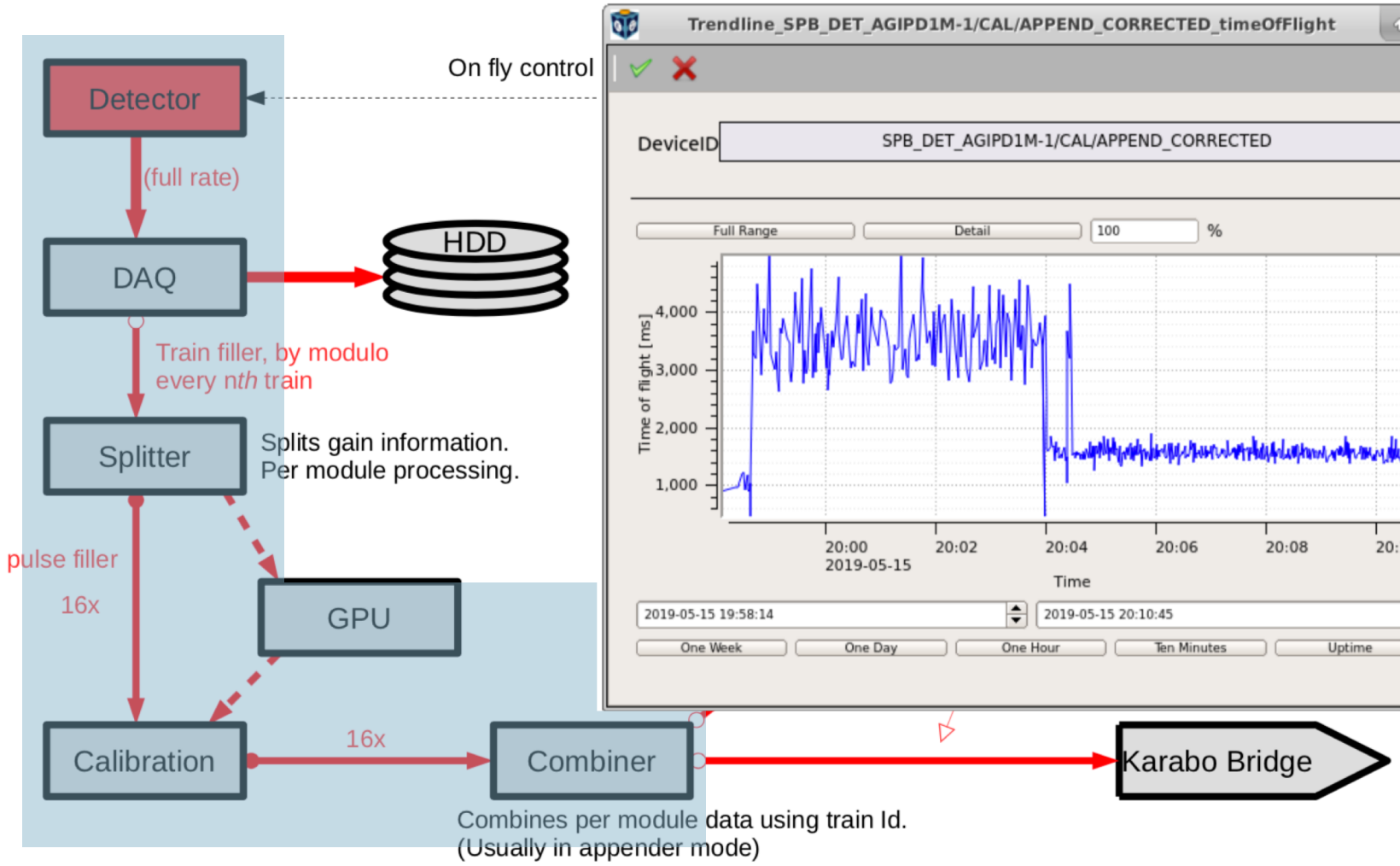
Data Solutions: Online Views – User Processing



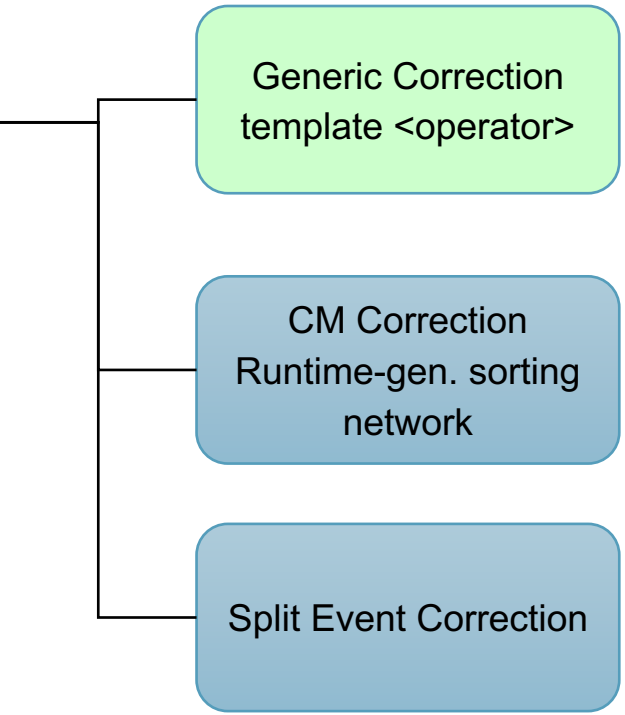
Data Solutions: Online Views – Facility Tools



Data Solutions: Online Views – Performance

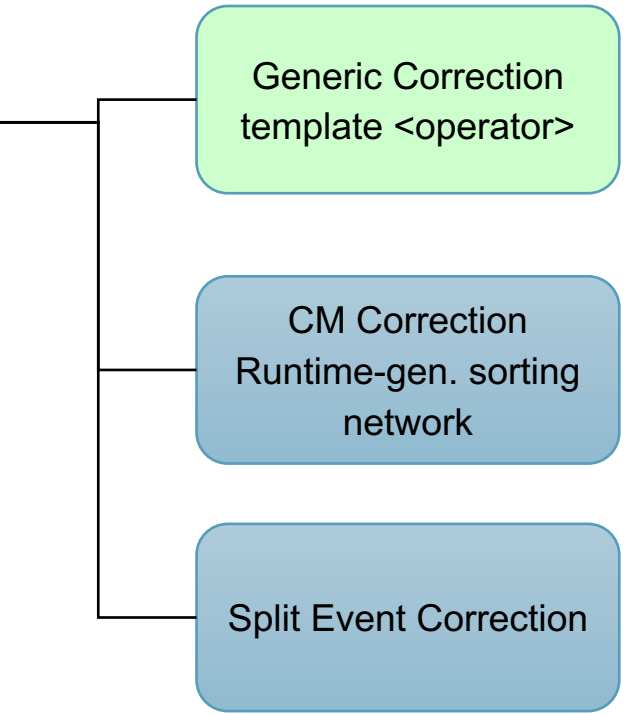


PyDetLib: Detector-related Data Processing




```
simpleCorrection = """
   #define TILE_DIM {{ tile_dim }}
   ...
   __global__ void simpleCorrection({{ data_type }}* __restrict__ a,
       const unsigned char* __restrict__ g,
       const {{ data_type }}* __restrict__ constant,
       const int* __restrict__ celltable){

       const int iX = threadIdx.x;
       const int iY = threadIdx.y;
       const int iZ = threadIdx.z;
       const int zStride = gridDim.z*STACK;
       const int yStride = gridDim.y*TILE_DIM;

       int idx = (blockIdx.z*STACK+iZ) + ...;

       if(idx < WIDTH*HEIGHT*zStride)
       {
           int cell = celltable[...];
           int idx_map = cell*GAINS + (iY+blockIdx.y*TILE_DIM) ...;
           if((bool)NO_GAINMAP){
               a[idx] {{ op }} constant[idx_map];
           } else {
               a[idx] {{ op }}  constant[idx_map+g[idx]];
           }
          __syncthreads();
       }
   }
"""
```

PyDetLib: Detector-related Data Processing




```
tpl = Template(kernels.simpleCorrection)
offsetCorrection = tpl.render(tile_dim=tile_dim, width=shape[0],
                              height=shape[1], stack=zDim,
                              mem_cells=memCells, gains=gainLen,
                              data_type='float',
                              no_cell_table=noCellTable,
                              no_gain_map=noGainMap,
                              op="-=")
```

PyDetLib: Detector-related Data Processing

Generic Correction template <operator>

CM Correction Runtime-gen. sorting network

Split Event Correction

karaboDevices › pyDetLib › **Repository**

577c9a6ee... | pyDetLib / .. / kernels / **simple_correction.py** | Find file | Blame | History | Permalink

Add GPU gain thresholding ...
Steffen Hauf committed a year ago — **577c9a6e**

simple_correction.py 4.6 KB — Edit | Replace | Delete

```
"""
An simple correction kernel
"""
simpleCorrection = """
    #define TILE_DIM {{ tile_dim }}
    #define WIDTH {{ width }}
    #define HEIGHT {{ height }}
    #define STACK {{ stack }}
    #define MEMCELLS {{ mem_cells }}
    #define GAINS {{ gains }}
    #define NO_CELLTABLE {{ no_cell_table }}
    #define NO_GAINMAP {{ no_gain_map }}

    __global__ void simpleCorrection({{ data_type }}* __restrict__ a, const unsigned char* __restrict__ g,

        const int iX = threadIdx.x;
        const int iY = threadIdx.y;
        const int iZ = threadIdx.z;
        const int zStride = gridDim.z*STACK;
        const int yStride = gridDim.y*TILE_DIM;

        int idx = (blockIdx.z*STACK+iZ) + (iY+blockIdx.y*TILE_DIM)*zStride + (iX+blockIdx.x*TILE_DIM)*zStri


        if(idx < WIDTH*HEIGHT*zStride)
        {
            //case where celltable == NULL
            //no need for shared memory as zindexing will be coalesced
            if((bool)NO_CELLTABLE){
                int cell = (STACK*blockIdx.z+threadIdx.z) % MEMCELLS;
                int idx_map = cell*GAINS + (iY+blockIdx.y*TILE_DIM)*MEMCELLS*GAINS + (iX+blockIdx.x*TILE_DI
                if((bool)NO_GAINMAP){
                    a[idx] {{ op }} constant[idx_map];
                } else {
                    a[idx] {{ op }} constant[idx_map+g[idx]];
                }



            } else {
                //random access to the constant may occur, it should be loaded into shared memory

                __shared__ {{ data_type }} tile[TILE_DIM][TILE_DIM][MEMCELLS+1][GAINS];
                //x and y will fit to the pixels to correct for but need to limit or extend in z direction
                int zDiff = MEMCELLS - blockDim.z;
```

PyDetLib: Detector-related Data Processing

- Generic Correction template <operator>
- CM Correction Runtime-gen. sorting network
- Split Event Correction

karaboDevices › pyDetLib › **Repository**

577c9a6ee... | pyDetLib / .. / kernels / **simple_correction.py** | Find file | Blame | History | Permalink

Add GPU gain thresholding ...
Steffen Hauf committed a year ago — **577c9a6e**

simple_correction.py 4.6 KB — Edit | Replace | Delete

```
"""
An simple correction kernel
"""
simpleCorrection = """
    #define TILE_DIM {{ tile_dim }}
    #define WIDTH {{ width }}
    #define HEIGHT {{ height }}
    #define STACK {{ stack }}
    #define MEMCELLS {{ mem_cells }}
    #define GAINS {{ gains }}
    #de
    #de

    __g                                                                    ict__ g,

                                                                           IM)*zStri

                int cell = (STACK*blockIdx.z+threadIdx.z) % MEMCELLS;
                int idx_map = cell*GAINS + (iY+blockIdx.y*TILE_DIM)*MEMCELLS*GAINS + (iX+blockIdx.x*TILE_DI
                if((bool)NO_GAINMAP){
                    a[idx] {{ op }} constant[idx_map];
                } else {
                    a[idx] {{ op }} constant[idx_map+g[idx]];
                }



            } else {
                //random access to the constant may occur, it should be loaded into shared memory

                __shared__ {{ data_type }} tile[TILE_DIM][TILE_DIM][MEMCELLS+1][GAINS];
                //x and y will fit to the pixels to correct for but need to limit or extend in z direction
                int zDiff = MEMCELLS - blockDim.z;
```

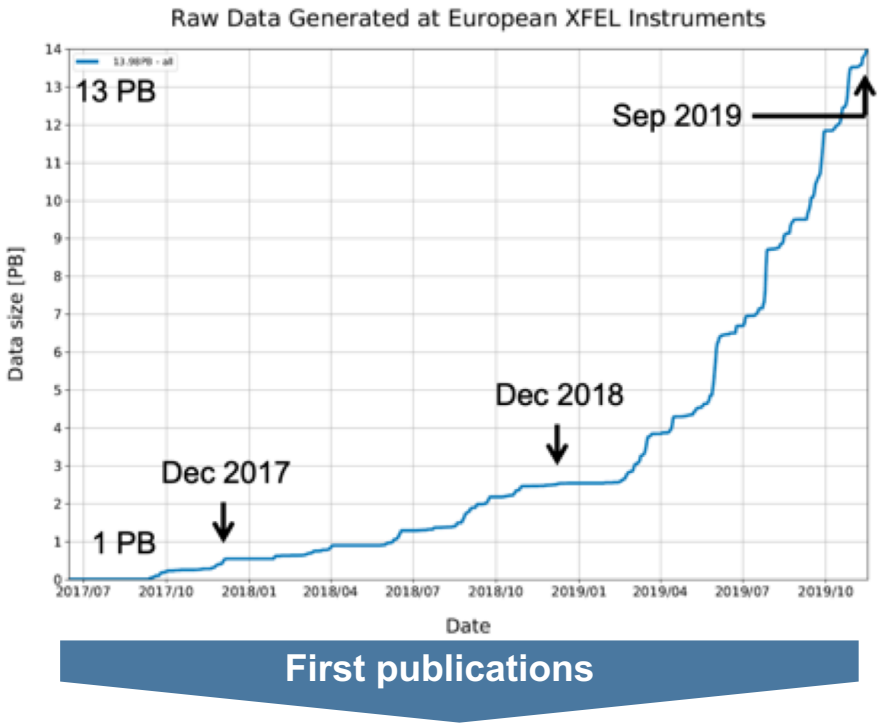
With Volta, global memory through-put increased such that it GM access on large data chunks became faster than small-chunked shared memory access

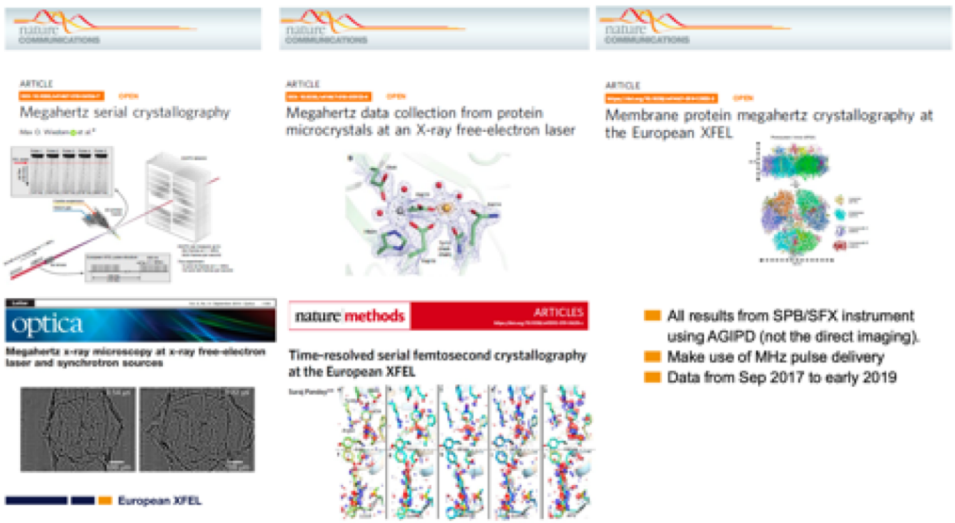
→ Code reduction by factor 10!

Data Challenges and Lessons Learned

Challenges: Data Reduction

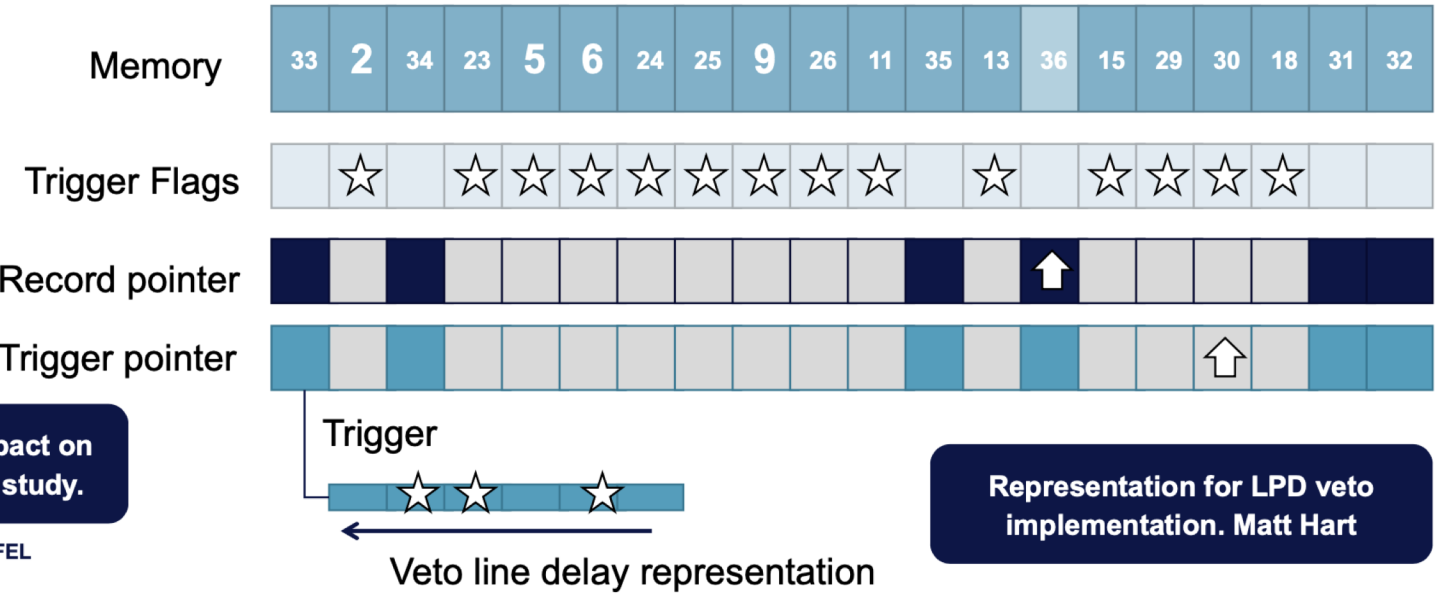
- When running smoothly, single experiments can produce ~1PB per week
  - Not all data contains sample hits
  - Data is reduced during analysis, but not upon storage
- Mid- and long-term running XFEL is only feasible if data is reduced as early as possible
  - More efficient storage
  - Faster analysis on actually interesting data
  - Better online monitoring
  - Requires very good understanding of our detectors





Challenges: Vetoing on Detector is not a Reduction Solution on its own

- Designed to reuse memory cells during a train, i.e. increase the yield of meaning data
- No need to reject data within micro seconds with less data, when you could **do it with more data, later**
- Data reduction using the Veto would only become necessary to reduce the data load on the DAQ, though is this currently not the problem (exception 4M AGIPD)



Use of Veto has shown impact on calibration. Needs further study.

Representation for LPD veto implementation. Matt Hart

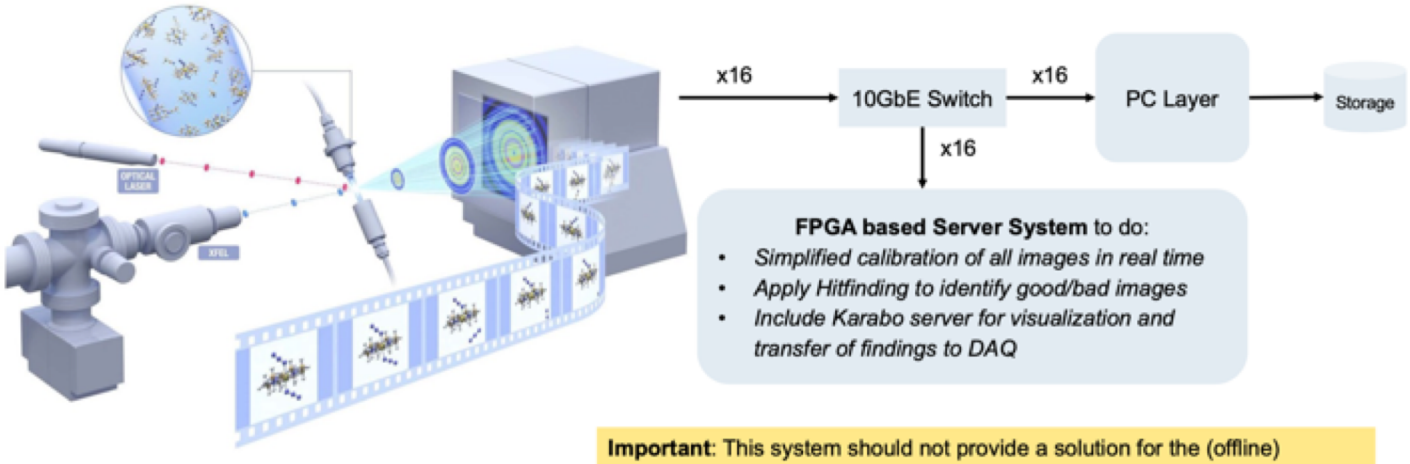
- At 2700 pulses nominal, XFEL.EU oversamples the detector memory
- Not less data, but same amount of data with more useful content
- **To use for reduction, the amount of useful content needs to be fixed**

European XFEL

Challenges: Context Specific Reductions

- First step, filtering known empty pulses
  - From detector characterization observations: prefer in software, not using VETO, as complicates calibration
- Solutions which take context into account, e.g. experiment is interested in Bragg peaks
  - Software or FPGA solutions are being investigated
  - Start with only tagging data, then get user feedback on our filters
  - Final solution: filter most, but pass through some data which would otherwise be filtered for verification purposes

Planned demonstrator project for AGIPD at SPB



Important: This system should not provide a solution for the (offline) calibration and processing with highest accuracy. It should provide a fast and low-latency online monitoring solution to provide valuable feedback to the users in real-time! It could also provide classification information of the images for later decisions for processing or data reduction.

Challenges: Context Specific Reductions

- First step, filtering known empty pulses
  - From detector characterization observations: prefer in software, not using VETO, as complicates calibration
- Solutions which take context into account, e.g. experiment is interested in Bragg peaks
  - Software or FPGA solutions are being investigated
  - Start with only tagging data, then get user feedback on our filters
  - Final solution: filter most, but pass through some data which would otherwise be filtered for verification purposes
- Other facilities face the same challenges:
  - LCLSII mandatory reduction by factor 10x early on
    - ► Users can provide algorithm specific to experiment
    - ► Facility “hard“ filter will take care of anything above 10% input if the user algorithm didn‘t
    - ► XFEL is looking into collaboration here, e.g. in the context of HIR3X
  - Multiple „power“ user groups have experience and knowledge that can help

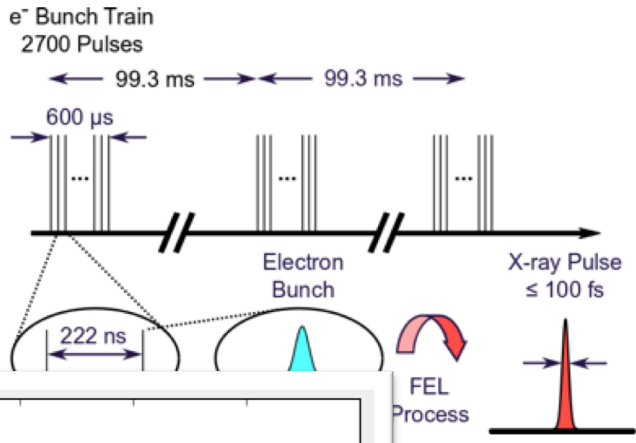
Challenges: Technical vs. Procedural

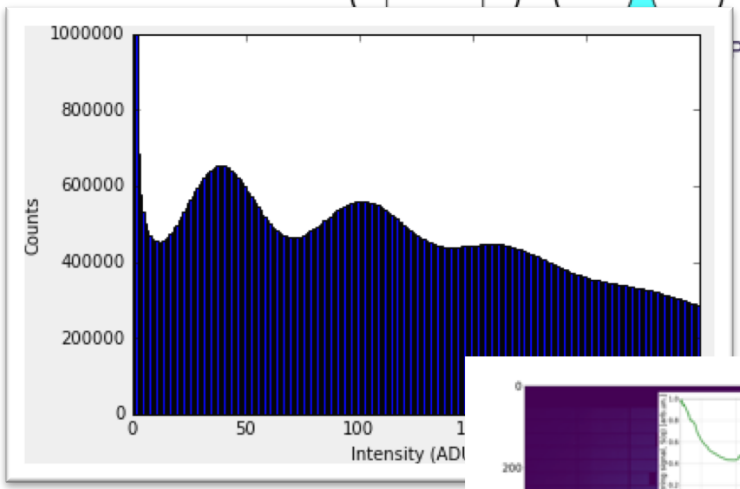
Filtering „dark“ data
Straight forward, but needs to be configurable on pattern Reduction: 0%-99% depending on scenario.

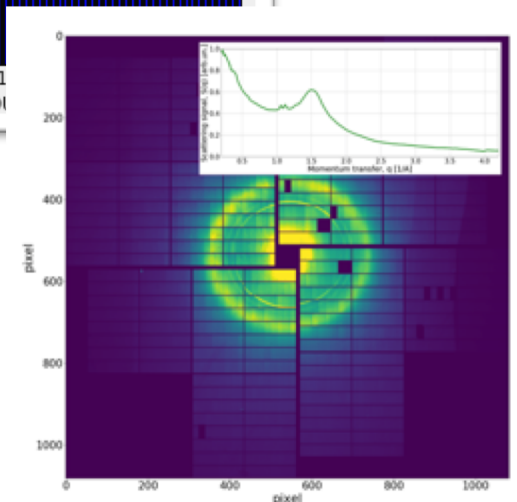
Compression
Will not gain much for data containing white noise (< ~10%), good for gain maps or bad pixel maps (> 90%).

Event filtering
Requires good calibration, can be done e.g also using zero-suppression and compression: ~0-99% depending on application.

Dimension reduction
E.g. azimuthal integration: ~1000x reduction if going from 2D(3D) to 1D(2D).







Challenges: Technical vs. Procedural

Filtering „dark“ data
Straight forward, but needs to be configurable on pattern Reduction: 0%-99% depending on scenario.

Compression
Will not gain much for data containing white noise (< ~10%), good for gain maps or bad pixel maps (> 90%).

Event filtering
Requires good calibration, can be done e.g also using zero-suppression and compression: ~0-99% depending on application.

Dimension reduction
E.g. azimuthal integration: ~1000x reduction if going from 2D(3D) to 1D(2D).

Experiment Specifics
Facility side knowledge of pulse pattern
User configuration option of what is important aside from FEL pulses.

Acknowledgement of Reduction Need
In contrast to frequent previous experience at Synchrotrons or even other FELs where persisting “all“ data was possible

Identification of >90% use cases
Focus on most common experimental techniques and stream-line facility data pipelines: crystallography, pump-probe techniques and symmetry assumptions

Transparency and Validation
Need to convince our users that data rejection does not harm scientific outcome

Challenges: Data Reducion – what to gain?

Calibration

- ► Corrects data → 16 bit unsigned integer to 32 bit float values = +0.5 x data x 2
- ► Digitizes gain → 16 bit unsigned integer to 8 bit uint values = + 0.5 x data x 0.5
- ► Adds bad pixel mask → additional 32 bit unsigned int per image = + 0.5 x data x 2

data x 2.25

- HDF5 transparent inline compression using lossless LZF algorithms
  - Gain and Bad Pixels data contain very few distinct values and large regions of the same values, e.g. zeros, blocks of bad ADCs etc, and thus compress efficiently
  - Fast algorithm: measured 5% speed penalty
    - ► **Data x ~1.0**
  - **Zeroing image data below n sigma noise:** not lossless anymore, < **Data x 1**

Approach	RAW	No Compression	Compression (LZF)	+ zeroing (3 sigma)	+ zeroing (3 sigma), different run	+ zeroing (10 sigma),	+ static bad pixel map
File size	4.1G	8.9G	2.9G-4.0GB	2.8G	2.0G-2.9G	0.8-2.7G	2.7G
Relative	100%	220%	**70-100%**	**68%**	**50-70%**	**20-65%**	66%

reduction is very much data-dependent

Summary: Data lessons learned in early user operation

- **Well running experiments can produce up to 1PB of data in a single beamtime**
  - **Since we are a user facility, how such an experiment may look like can change on a daily basis** → contrasts to particle physics.
  - However, a flexible toolkit of reduction options might cover many experimental use cases

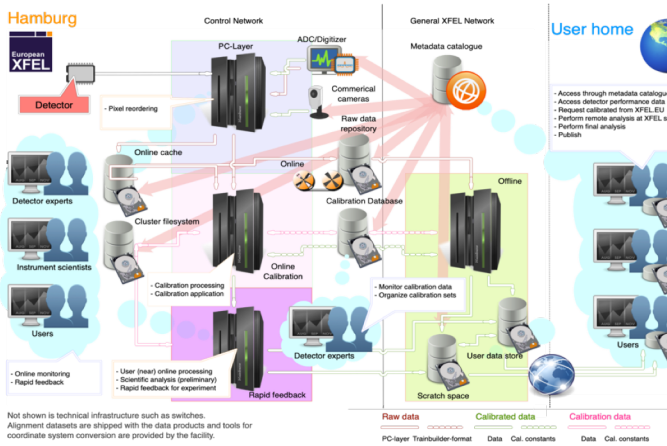
- Reducing this data in an efficient and appropriate way is one of the key challenges
  - **Users will not care so much about our storage costs but be interested in maximing useful feedback and speeding up analysis**
  - Frequently, users have synchrotron experience where data rates in the past have not been such an issue. There is a “pyschological“ componenent in raising awareness into the challenges that facilities like XFEL entail.
  - **We observe two user categories:**
    - ► **Power users** who are accustomed to handling such data amounts and often are willing to contribute with solutions as they understand the problems
    - ► **Non-power users**, who ideally only want a very reduced data product. Getting there should be as transparent to them as possible.

Summary: Data lessons learned in early user operation

- Software:
  - **Python** is by far most common nowadays. Most of the faclity analysis code, calibration processing and data interfaces are written in it
  - **C++**: still a serious player, used in DAQ and Control – where tight control over data types, referencing and concurrency is needed. Also some user code relies on this
  - Matlab: frequently required to run user analysis propagated from synchrotron environments, but more an more people moving to Python
- Hardware:
  - For XFEL data with large detectors, **memory is key**: Maxwell nodes have up to >700GB, standard configuration 512GB, fast IO (GPFS) is mandatory
  - GPUs: used in online processing and some user codes, but we also observe I/O rather than processing limitations → efficient data reduction might change this
  - FPGAs: becoming more important. XFEL hast started a project for high-level programming there. Recent developments of synthesizing Python and other HLL code directly onto FPGA should be highly interesting to us as the work well with frequent user turnaround.

Summary: Data lessons learned in early user operation

- Not mentioned (in Detail):
  - Electronic log books: currently using the PSI elog, but many requests for tighter integration with our data producers
  - Reproducability
  - Open Access after embargo period → how to make sure data useful for those who didn‘t participate in experiment
  - Collaborative software development and code review: git + gitlab, readthedocs, Docker, ...



- **Software developments are getting more and more complex:**
  - **A great scientist is not automatically a good software developer**
  - **Communication: scientists „fear“ written requirements, IT people crave them**
  - **Flexible, fast and I want to be able to change it → usually will not work**
  - **Users are “app-pampered“: sometimes the complex system still has to be understood, but also does it always have to be that way?**
  - **Do you really want the raw data?**

≠

Outlook

- More detectors will be installed at the European XFEL in the next months and years
  - AGIPD4M
  - HiBEF AGIPD
  - Second DSSC (DePFET)
  - More Jungfrau detectors
- Next generation of detectors is in definition phase. Wishlist includes:
  - More and smaller pixels
  - More memory cells
  - CW operation
  - Generally, more data
- The need for efficient data management, provenence and reduction will become even more pressing
  - Other user facilities are facing similiar challenges: LCLS II, many synchrotrons
  - Internal R&D projects have been started: FPGA, Software filtering
  - There are chances for collaboration
    - ► Among facilities
    - ► With user groups

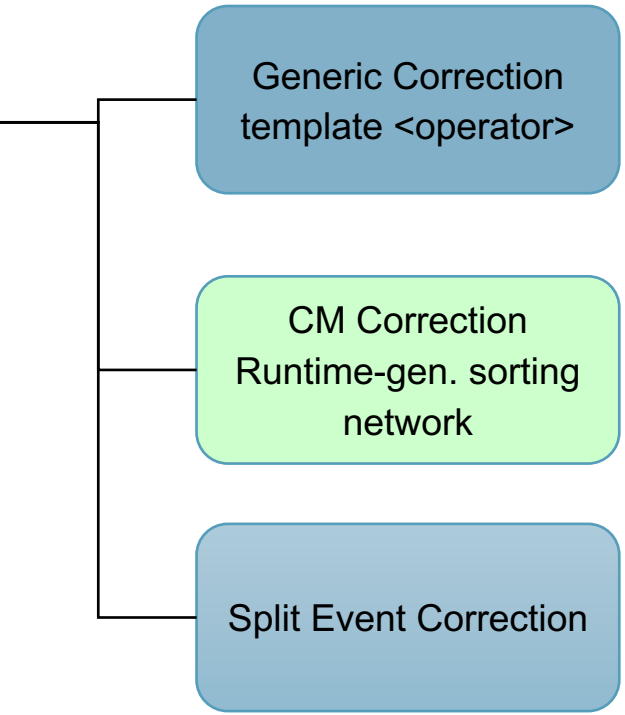
Outlook

- More detectors will be installed at the European XFEL in the next months and years
  - AGIPD4M
  - HiBEF AGIPD
  - Second DSSC (DePFET)
  - More Jungfrau detectors
- Next generation of detectors is in definition phase. Wishlist includes:
  - More and smaller pixels
  - More memory cells
  - CW operation
  - Generally, more data
- The need for efficient data management, provenence and reduction will become even more pressing
  - Other user facilities are facing similiar challenges: LCLS II, many synchrotrons
  - Internal R&D projects have been started: FPGA, Software filtering
  - There are chances for collaboration
    - ► Among facilities
    - ► With user groups

Calibration & data reduction are high priority in the XFEL.EU data department

Backup

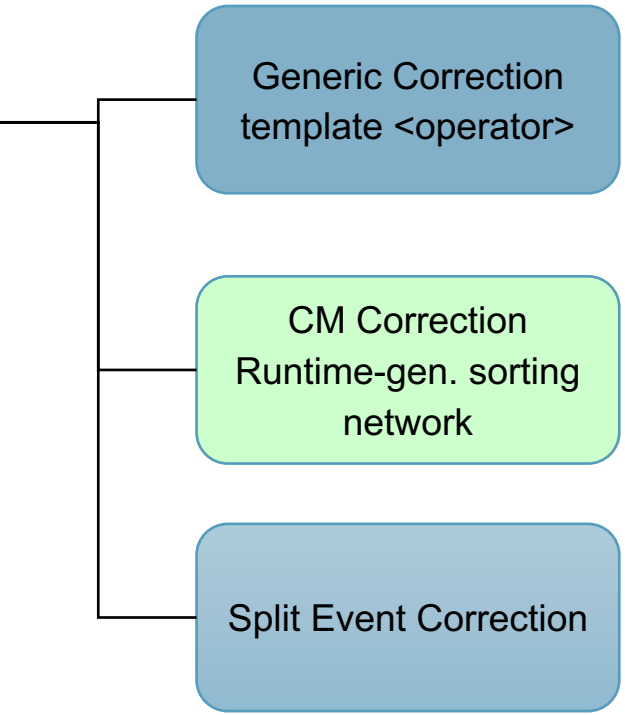
PyDetLib: Detector-related Data Processing




```
def oddeven_merge_sort_range(lo, hi):
    if (hi - lo) >= 1:
       mid = lo + ((hi - lo) // 2)
       for i in oddeven_merge_sort_range(lo, mid):
           yield i
       for i in oddeven_merge_sort_range(mid + 1, hi):
           yield i
       for i in oddeven_merge(lo, hi, 1):
           yield i


def oddeven_merge_sort(length):
   """ "length" is the length of the list to be sorted.
   Returns a list of pairs of indices starting with 0 """
   for i in oddeven_merge_sort_range(0, length - 1):
       yield i
```

Python generates a near-optimal sorting network for variable input data sizes at run time

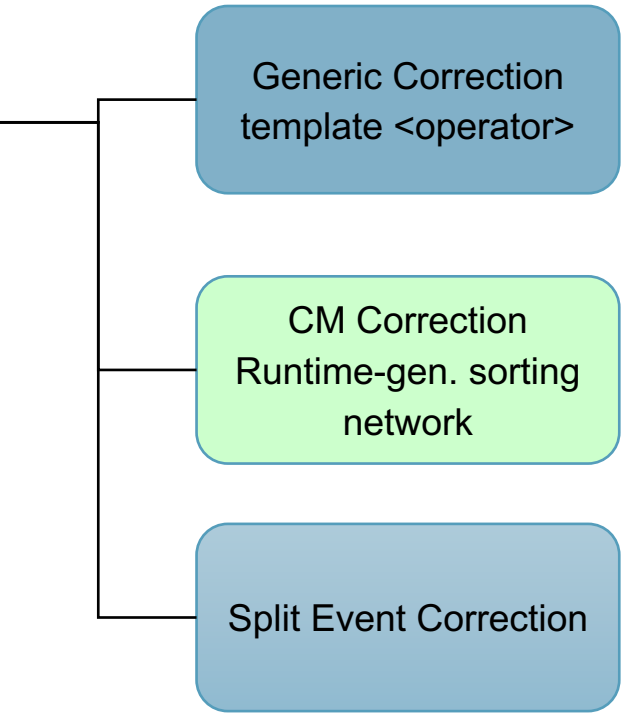
PyDetLib: Detector-related Data Processing




```
def generateNetwork(datalength, returnStats):
   pairs_to_compare = list(oddeven_merge_sort(datalength))
   ...
   for j, i in enumerate(pairs_to_compare):
       k = 0
       ...
       if len(k) > 0:

          code.append(
               "if (laneId < %d) if(a[c[%d][laneId][0]]...}".format(.

           if warpSize < 0:
               code.append("__syncthreads();")


def commonmode3dNetwork(datalength, returnStats=False):
   code, constant = generateNetwork(datalength, returnStats)
   code = code.replace('{{ specialization }}', """

       __syncthreads();

       #pragma unroll
       for(int i = 0; i < YSPAN; i++){
               a[idx+(iY*YSPAN+i+...] -= med[iX][iZ];

       } """
```

The use this network to generate CUDA-C code

PyDetLib: Detector-related Data Processing



Speedup for GPU algorithms (very old numbers!)

Detector (Pixels x mem. cells)	GPU (ms)	CPU (ms)**	Speed-Up
LPD (128x32x512)	**26.5**	3473	~130x
AGIPD (128x512x352)	**344.9**	15394	~45x
DSSC (512x128x512)	**477.4**	38456	~75x
LPD Supermodule (256x256x512)	**466.1**	35796	~75x