

Detectors and Calibration (Processing)

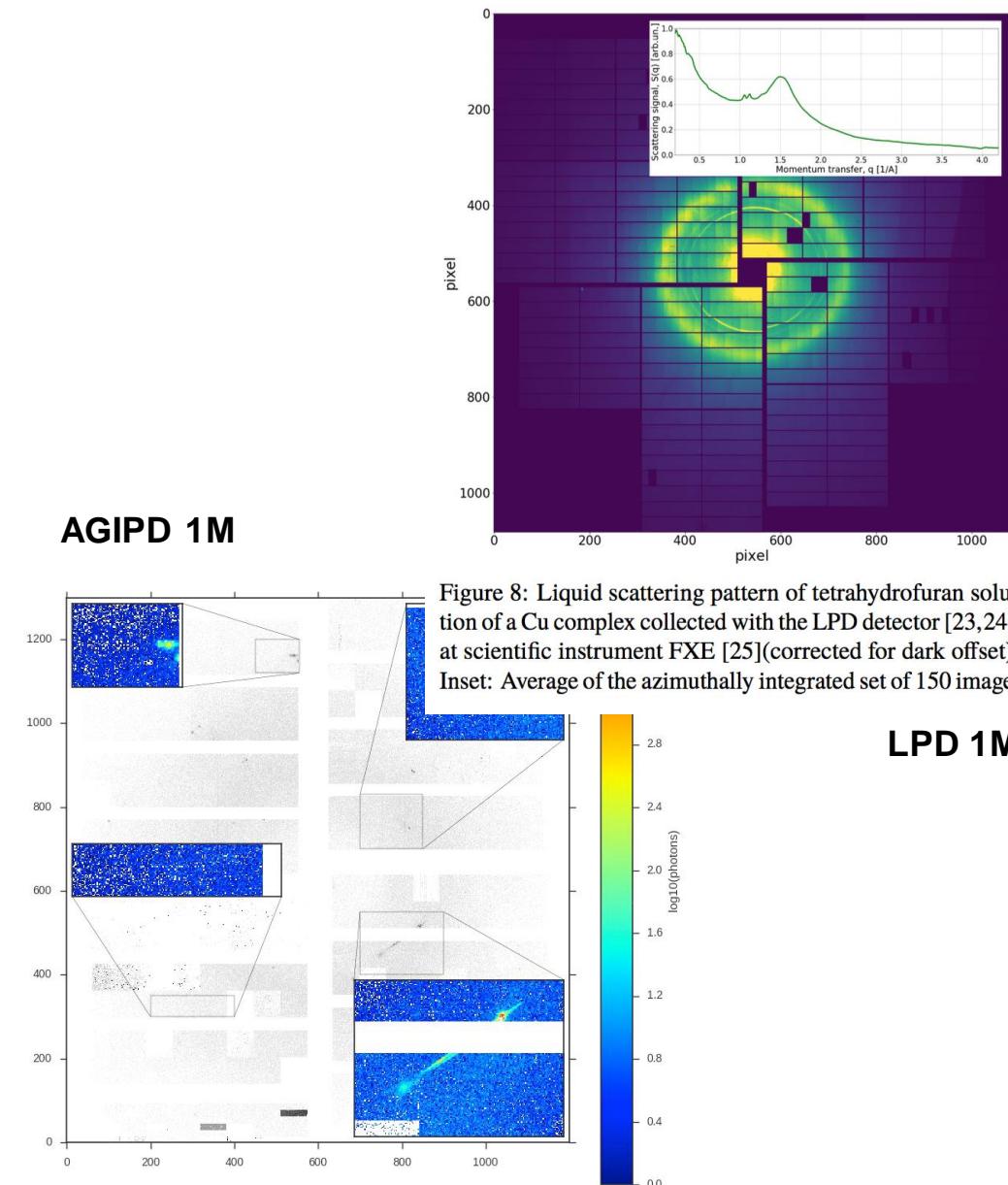


Data Analysis & Karabo Workshop, 23.01.2018

Steffen Hauf

Overview

- Detectors @ European XFEL
- Detector Calibration
 - Tools we use
 - Online Calibration
 - Offline Calibration
 - Characterization Processing
 - Documentation
 - Simulations
- With contributions from AGIPD, DSSC and LPD consortia, first users and instruments, DET, CAS, ITDM,
- 5 user beam times, 2 commissioning beam times + commissioning time → ~300 hours of data taking for both detectors.
- Detector up-time currently ~ 90%



Overview

- Detectors @ European XFEL
- Detector Calibration
 - Tools we use
 - Online Calibration
 - Offline Calibration
 - Characterization Processing
 - Documentation
 - Simulations
- With contributions from AGIPD, DSSC and LPD consortia, first users and instruments, DET, CAS, ITDM,
- 5 user beam times, 2 commissioning beam times + commissioning time → ~300 hours of data taking for both detectors.
- Detector up-time currently ~ 90%



LPD @ FXE

Experiment/proposal	Number of hours	Raw data size [TB]
p2016	60	4.1
p2073	60	2.3
p2052	60	3.9
p2072	48	20
p2050	60	27
Total	288 h	57 TB

AGIPD @ SPB

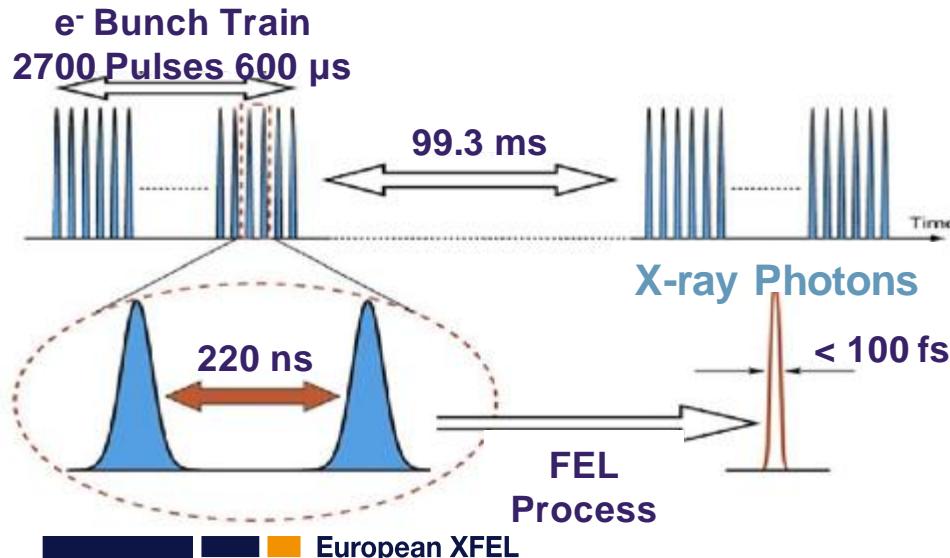
Experiment/proposal	Number of hours	Raw data size [TB]
p2012	60	57
p2042	60	87
p2017	60	49
p2066	48	80
p2013	60	116
Total	288 h	389 TB*

* Double size images, X-rays in every 2nd image

Detectors: from First Ideas to User Operation – 2006 til 2017

- The European XFEL pulse structure poses strict constraints on detectors (e.g. intensity and time structure)
- No commercial imaging detectors available
- Call for expression of interest launched in 2006
- 3 project proposals were selected with the goal to finally have at least one fast 2D imaging detector

e⁻ Bunch and X-ray Pulse Structure



European XFEL Project Team
an Deutsches Elektronen-Synchrotron DESY
in der Helmholtz-Gemeinschaft,
Notkestraße 85,
D-22607 Hamburg, Germany



Call by the:

European Project Team for the
X-ray Free-Electron Laser

for:

Expressions of Interest

to:

Develop and Deliver
Large Area Pixellated X-ray
Detectors.

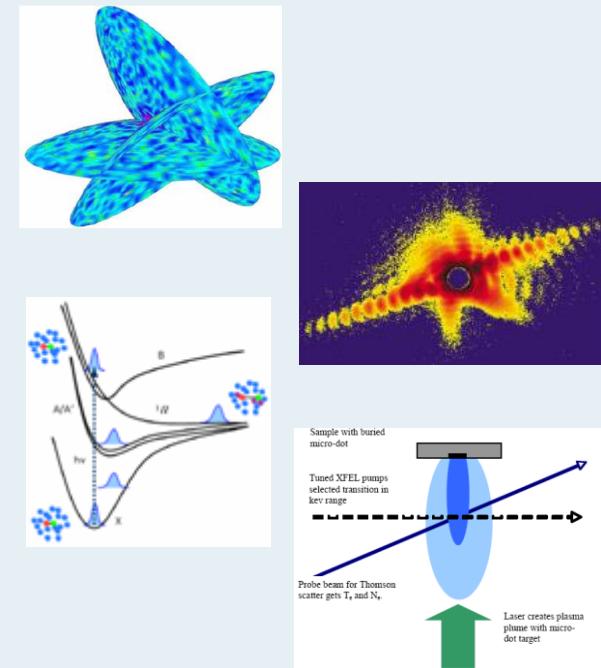
Deadline: 30 September 2006
<http://xfel.desy.de/xfelhomepage>

- Selected proposals
 - Adaptive Gain Integrating Pixel Detector
 - Large Pixel Detector
 - DEPFET Sensor with Signal Compression

XFEL Scientific Instruments

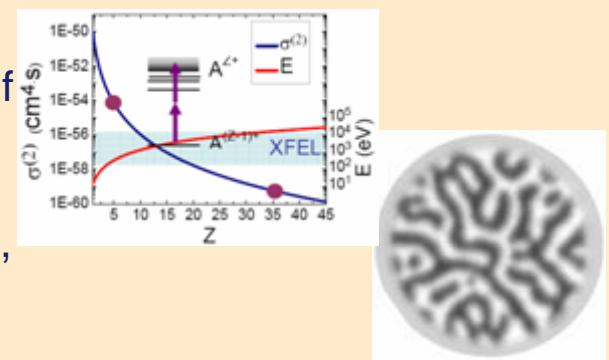
Hard X-Rays

- 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

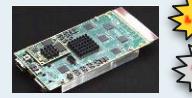
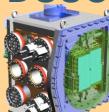
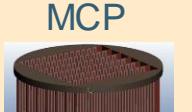


Soft X-Rays

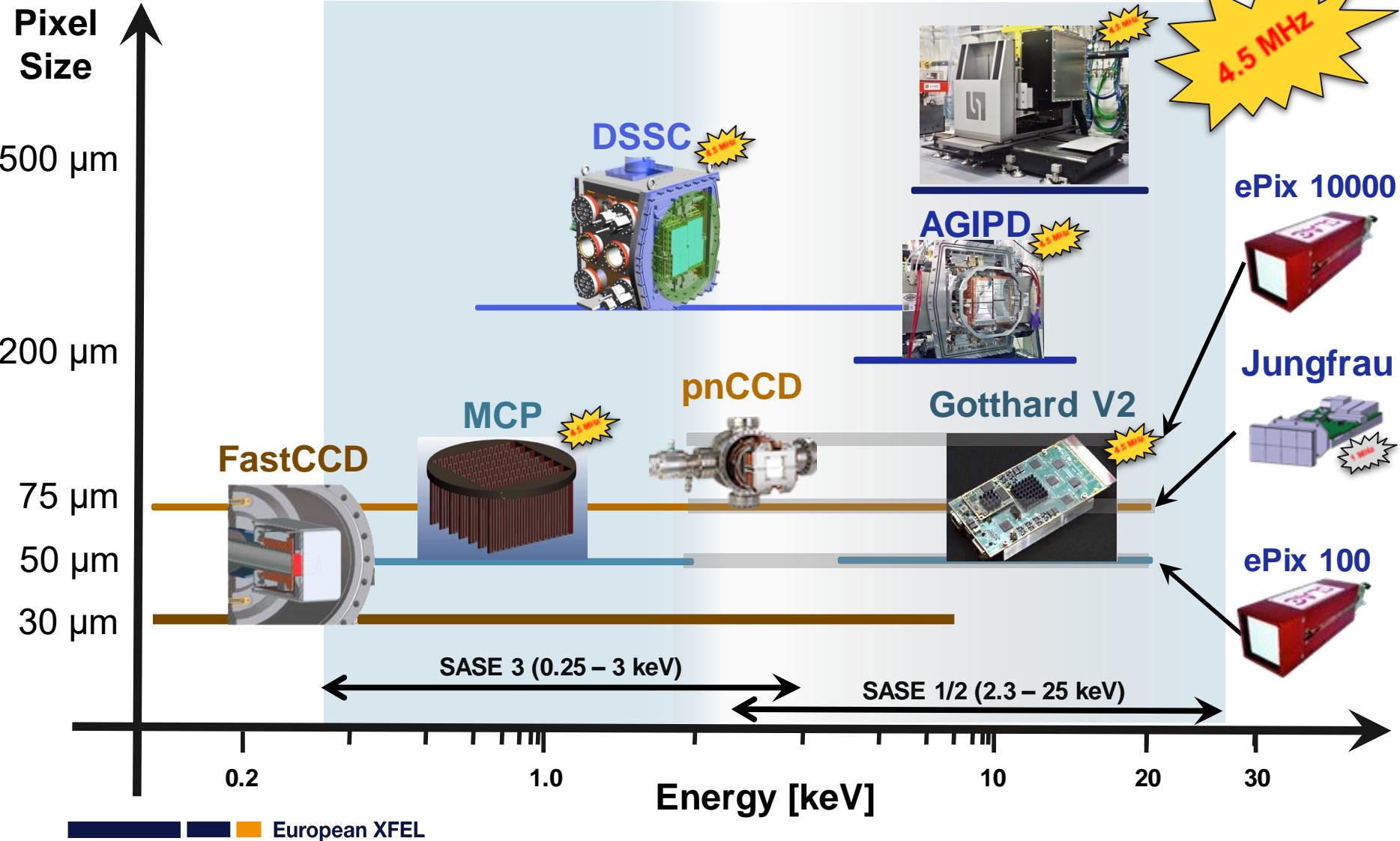
- SQS** **Small Quantum Systems**
Will examine the quantum mechanical properties of atoms and molecules.
- SCS** **Soft X-Ray Coherent Scattering/Spectroscopy**
Will determine the structure and properties of large, complex molecules and nano-sized structures.



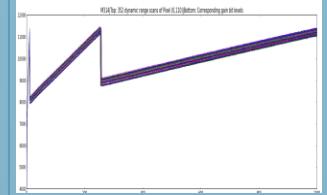
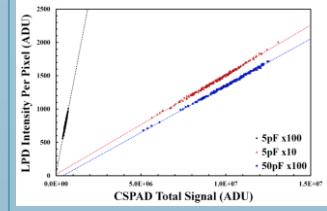
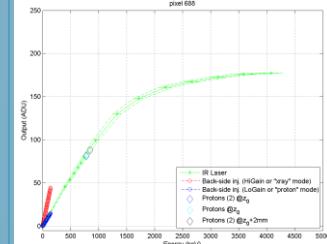
Detectors for the Scientific Instruments

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

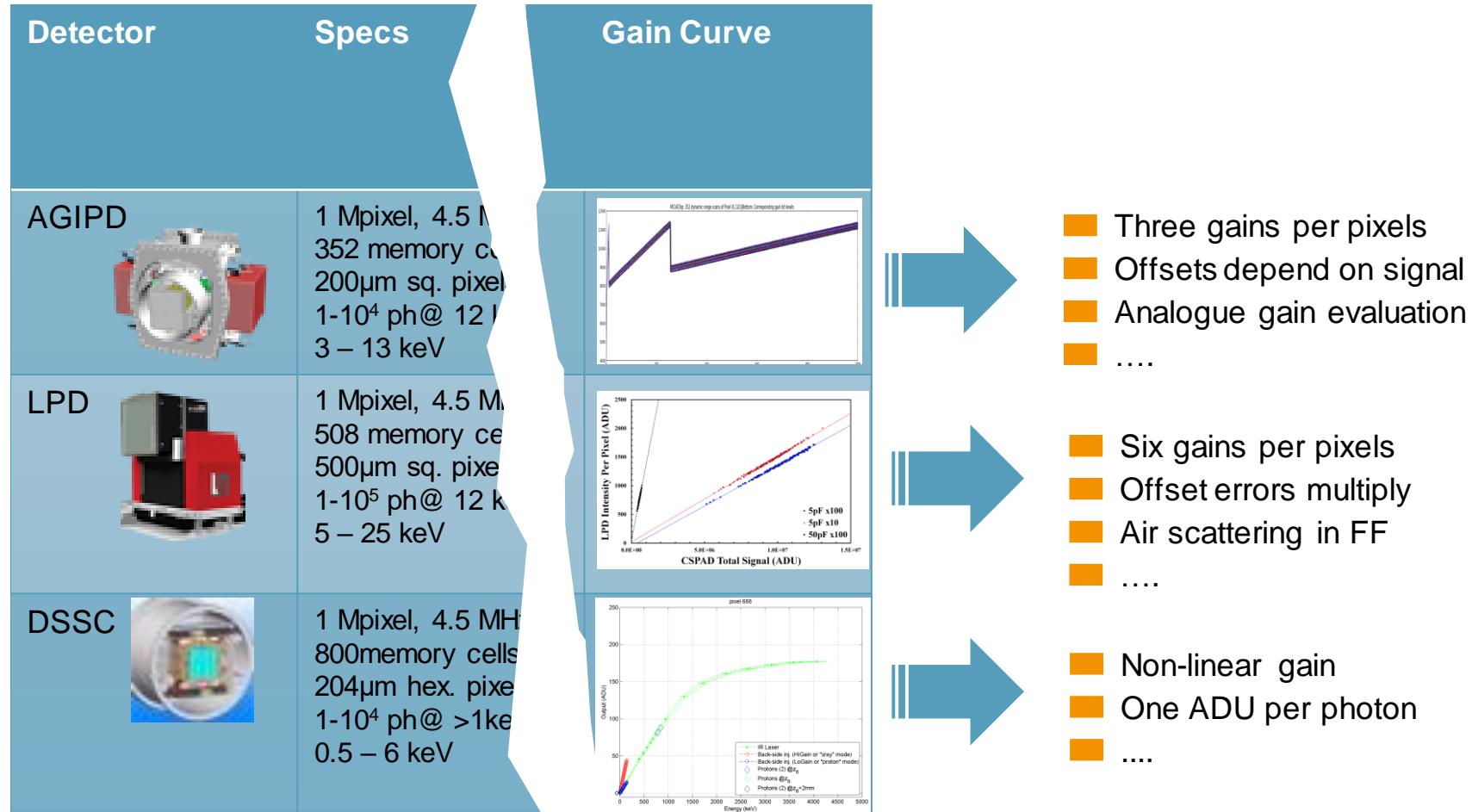
Detectors for the European XFEL



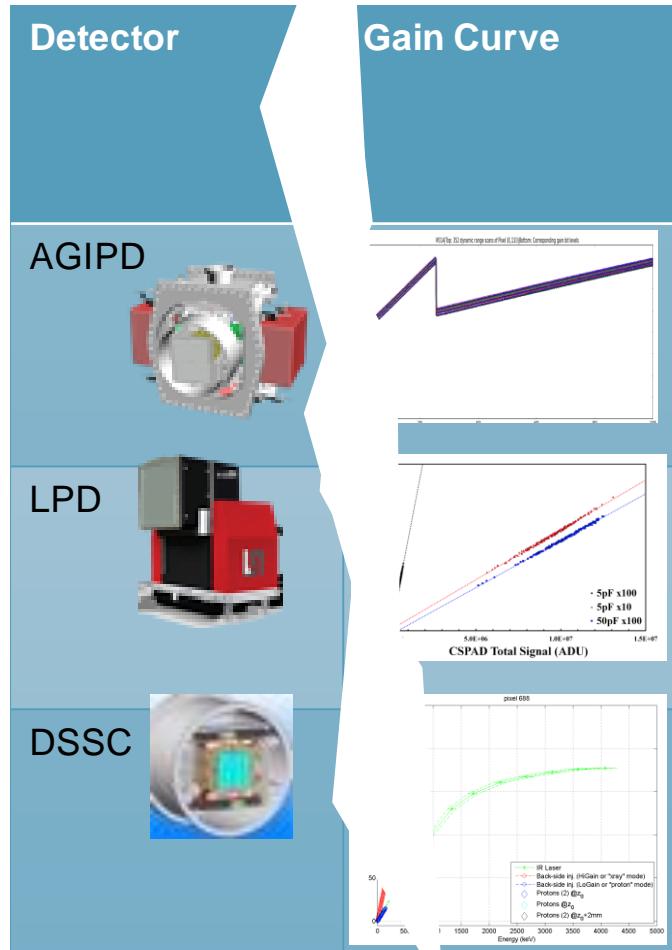
MHz Rate, High Dynamic Range Detectors – Challenges for Calibration

Detector	Specs	Modularity	Gain Switching	Gain Curve						
AGIPD	 <p>1 Mpixel, 4.5 MHz 352 memory cells 200µm sq. pixels 1-10⁴ ph@ 12 keV 3 – 13 keV</p>	16 modules in 2 cols x 8 rows on 4 quadrants	3 gain stages with automatic switching	 <p>AGIPD by [redacted] dynamic range steps of Pixel 0, 120 (bottom). Comparing gain to noise.</p>						
LPD	 <p>1 Mpixel, 4.5 MHz 508 memory cells 500µm sq. pixels 1-10⁵ ph@ 12 keV 5 – 25 keV</p>	16 modules per supermodule (2x8) 16 SM on 4 quadrants	3 gain stages with on front-end selection	 <p>LPD Intensity Per Pixel (ADU) vs CSPAD Total Signal (ADU)</p> <ul style="list-style-type: none"> - SpF x100 - SpF x10 - 50pF x100 						
DSSC	 <p>1 Mpixel, 4.5 MHz 800 memory cells 204µm hex. pixels 1-10⁴ ph@ >1keV 0.5 – 6 keV</p>	16 modules in 2 cols x 8 rows on 4 quadrants	Non-linear gain in ASIC (miniSDD), in sensor (DePFET)	 <p>pixel 685</p> <table border="1"> <tr> <td>IR Laser</td> <td>Back-side inj (HiGain or "base" mode)</td> </tr> <tr> <td>Protons D @ 1g</td> <td>Back-side inj (LoGain or "proton" mode)</td> </tr> <tr> <td>Protons D @ 1g 2mm</td> <td>Protons D @ 1g 2mm</td> </tr> </table> <p>Energy (keV)</p>	IR Laser	Back-side inj (HiGain or "base" mode)	Protons D @ 1g	Back-side inj (LoGain or "proton" mode)	Protons D @ 1g 2mm	Protons D @ 1g 2mm
IR Laser	Back-side inj (HiGain or "base" mode)									
Protons D @ 1g	Back-side inj (LoGain or "proton" mode)									
Protons D @ 1g 2mm	Protons D @ 1g 2mm									

MHz Rate, High Dynamic Range Detectors – Challenges for Calibration



MHz Rate, High Dynamic Range Detectors – Challenges for Calibration



- Three gains per pixels
- Offsets depend on signal
- Analogue gain evaluation
-

- Six gains per pixels
- Offset errors multiply
- Air scattering in FF
-

- Non-linear gain
- One ADU per photon
-

Calibration is non-trivial!

Facility-side calibration recommended

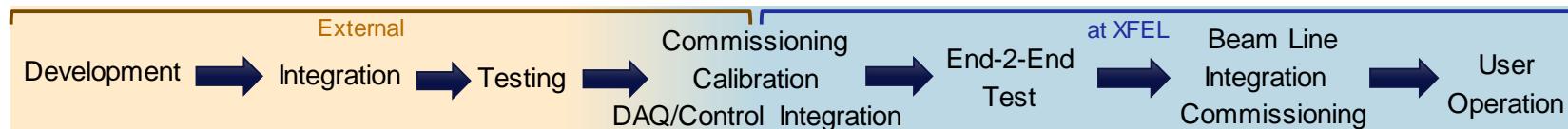
Detectors – Timeline and Status

Detector System	Beam Line	Scientific Instrument	Project Status	Arrival at XFEL	Installation at Experiment	
AGIPD		SASE I	SPB	Calibration/User Operation	December 2016	August 2017
LPD		SASE I	FXE	Calibration/User Operation	February 2017	July 2017
FastCCD		SASE III	SCS	Commissioning/Calibration	May 2016	May 2018
AGIPD		SASE II	MID	Integration	March/April 2018	August 2018
Gotthard V2		SASE I-III	FXE/HED/MID/SPB/Diagnostics	Development	July 2018	August 2018
DSSC MiniSDD		SASE III	SCS	Integration	September 2018	November 2018
pnCCD		SASE III	SCS/SQS	Integration	July 2018	November 2018
MCP DLD		SASE III	SQS	Development	≈ First half of 2018	Approx. 1 year after delivery
DSSC DEPFET		SASE III	SCS/SQS	Development	January 2020	November 2020

External
 Development → Integration → Testing → Commissioning → End-2-End Test → at XFEL → Beam Line Integration Commissioning → User Operation

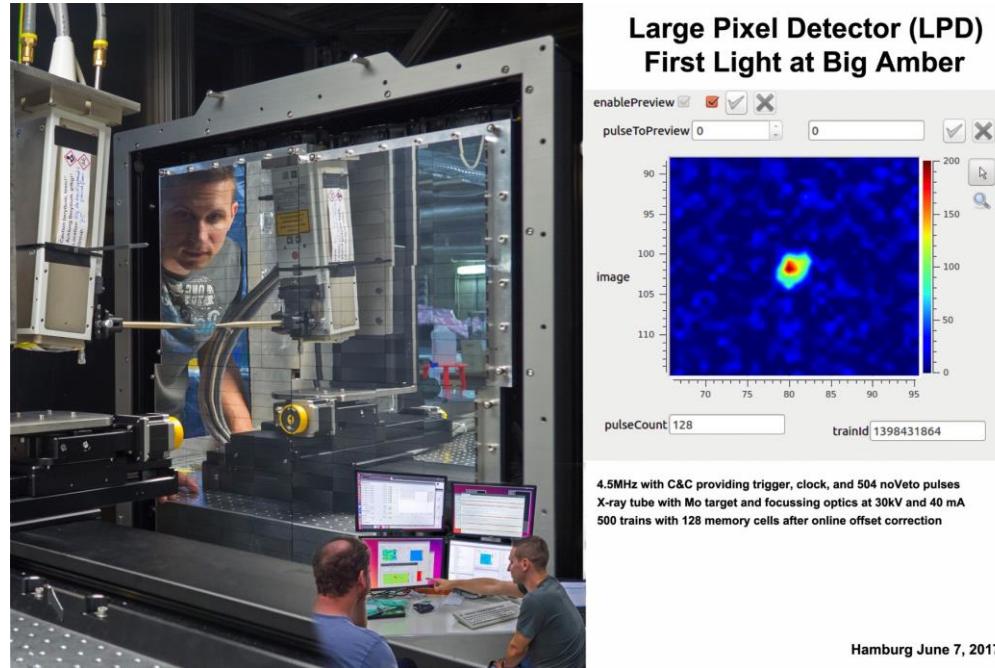
Detectors – Timeline and Status

Detector System	Beam Line	Scientific Instrument	Project Status	Arrival at XFEL	Installation at Experiment
ePix100 	SASE II	MID	Integration	February 2018	September 2018
ePix100 	SASE II	HED	Integration	February 2018	December 2018
Jungfrau 	SASE I	SPB	Development / Integration (1 mem cell)	April 2018	August 2018
Jungfrau 	SASE I	FXE	Development/ Integration (1 mem cell)	April 2018	August 2018
Jungfrau 	SASE II	MID	Development/ Integration (1 mem cell)	September 2018	October 2018
Jungfrau 	SASE II	HED	Development/ Integration (1 mem cell)	December 2018	January 2019



On the way to user operation – Detector End-to-End tests

- Status @ last XDAC: LPD and AGIPD in the HERA South lab, prepared for user operation.
- LPD and AGIPD transported from HERA South to Schenefeld in July 2017 (LPD) and middle of August (AGIPD)



- Much less time for E2E tests than originally planned. Some tests had to be postponed after the beam line installation.
- Inauguration event on September 1st: first images on AGIPD. **User operation started on September 14th!**

The AGIPD detector at SPB/SFX

Live demonstration experiment on the occasion of the European XFEL inauguration, including AGIPD data.
SPB/SFX Scientific Instrument
September 1st, 2017



European XFEL

LPD at FXE

- System worked stably
- Small problems with hardware were solved quickly (coolant leak, broken transceiver, etc.)
- Detector illuminated with X-rays at FXE for commissioning purposes
- Uses veto system to reject ‘empty’ images

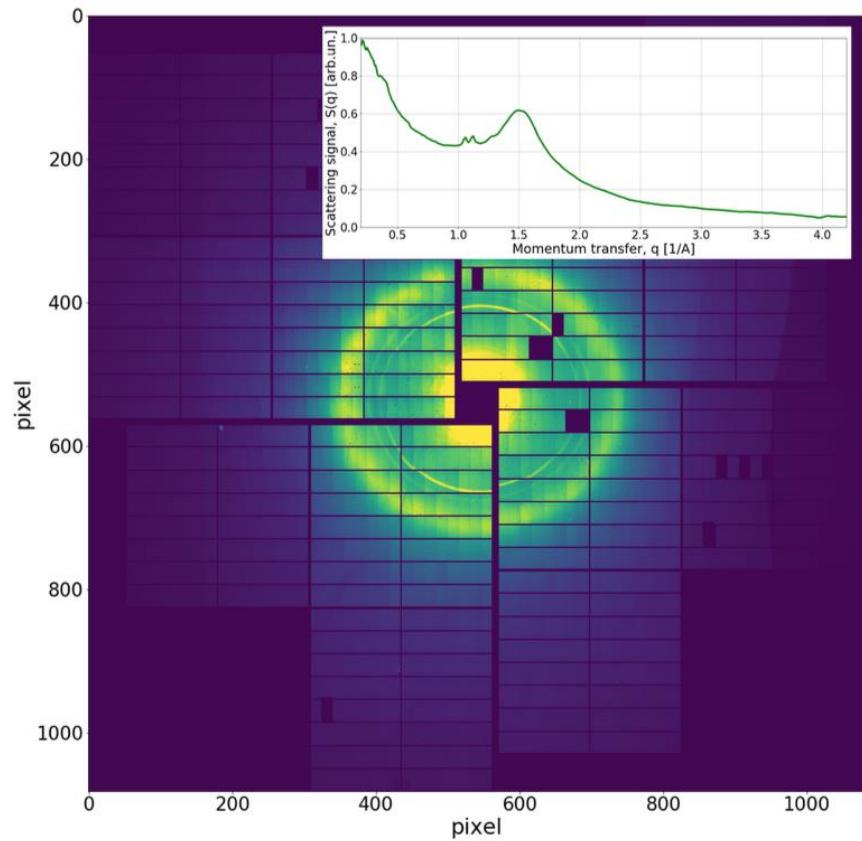


Figure 8: Liquid scattering pattern of tetrahydrofuran solution of a Cu complex collected with the LPD detector [23,24] at scientific instrument FXE [25](corrected for dark offset). Inset: Average of the azimuthally integrated set of 150 image.

LPD at FXE

- System worked stably
- Small problems with hardware were solved quickly (coolant leak, broken transceiver, etc.)
- Detector for correction of beam position
- Uses very little power

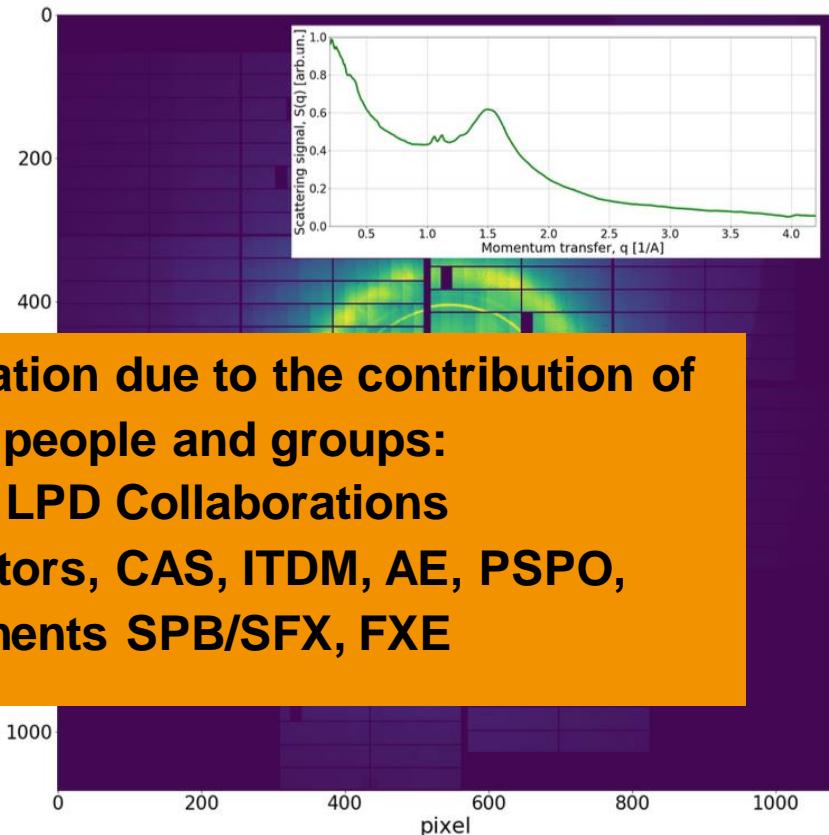
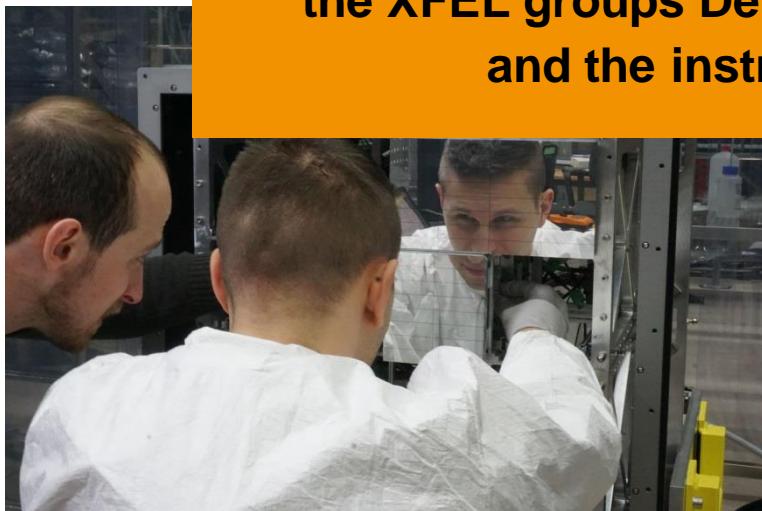


Figure 8: Liquid scattering pattern of tetrahydrofuran solution of a Cu complex collected with the LPD detector [23,24] at scientific instrument FXE [25](corrected for dark offset). Inset: Average of the azimuthally integrated set of 150 image.

Tools we use

Karabo – the European XFEL Control and Analysis Framework

- DAQ, online corrections, multi-module combination, and user exposure are all implemented in this framework. See e.g. Kuster et al.: *Detectors and Calibration Concept for the European XFEL*; Fangohr et al.: *Data Analysis Support at European XFEL*
- Detector Control is implemented in Karabo, working on top of consortium-provided libraries
- Karabo now follows a 2-4 week release schedule, with bug fixes made quickly available
 - ▶ Many improvements in responsiveness during last 6 months as mandated with ever growing installations
 - ▶ DAQ and calibration and detector control follow relevant updates in these release cycles, leading to stability and performance improvements
 - ▶ Beam-time like conditions are a must for proper evaluation

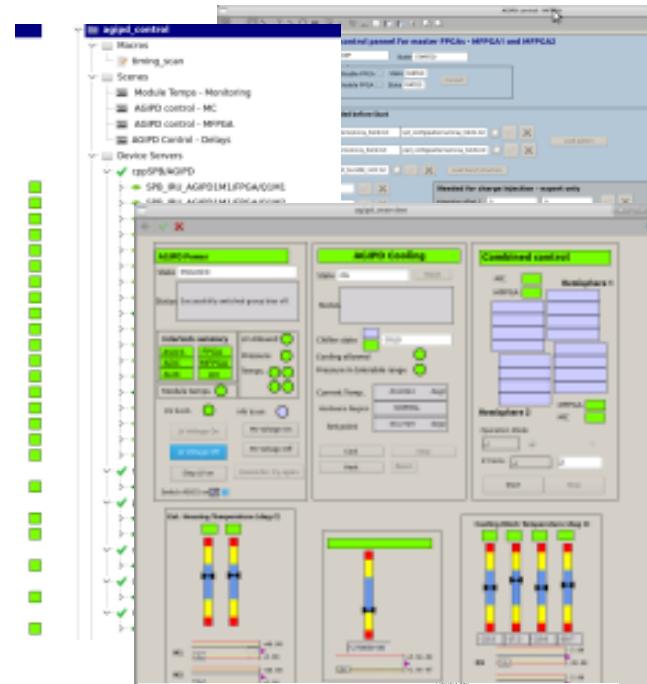
IPython and project Jupyter

- For offline analysis and characterization
- For transparency of what we do

ZMQ

- For exposing data to users via a well-established path

Karabo GUIs



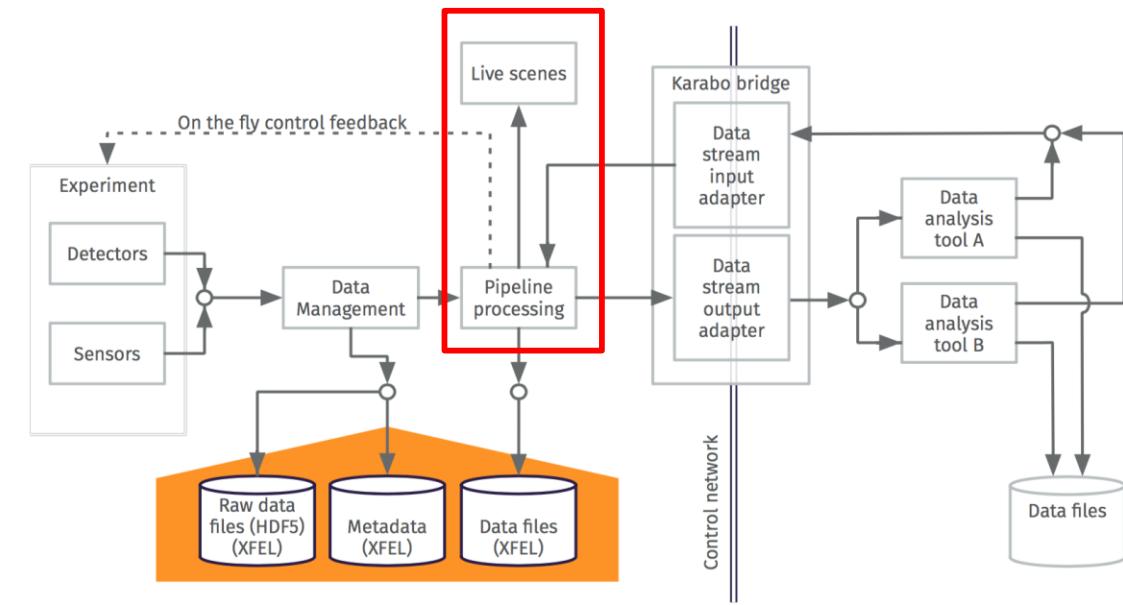
```
In [12]: # save everything to file.
for cap in capacitor_settings:
    run = [v for v in gains if run_items() if cap in v]
    file = '%s_offset_%s.h5'%(cap,out_folder)
    store_file = h5py.File(file,'w')
    for gm in offset_g[cap].keys():
        store_file['%s_offset/%s'%(cap,gm)] = offset_g[cap][gm]
    store_file['%s_offset/%s'%(cap,gm)] = noise_g[cap]
    store_file['%s_offset/%s'%(cap,gm)] = noise
    store_file.close()

In [13]: def show_hits(gain_to_preview, cap, ranges):
    res = Orderreddict()
    res[gain_to_preview] = offset_g[cap][gain_to_preview]
    res[gain_to_preview]['Offset'] = offset_g[cap][gain_to_preview]
    res[gain_to_preview]['Noise'] = noise_g[cap][gain_to_preview]
    res[gain_to_preview]['BadPixels'] = copy.copy(badpix_g[cap][gain_to_preview])
    res[gain_to_preview]['BadPixels'][res[gain_to_preview]['BadPixels'] == 0] = np.nan
    except: res[gain_to_preview] = None
```

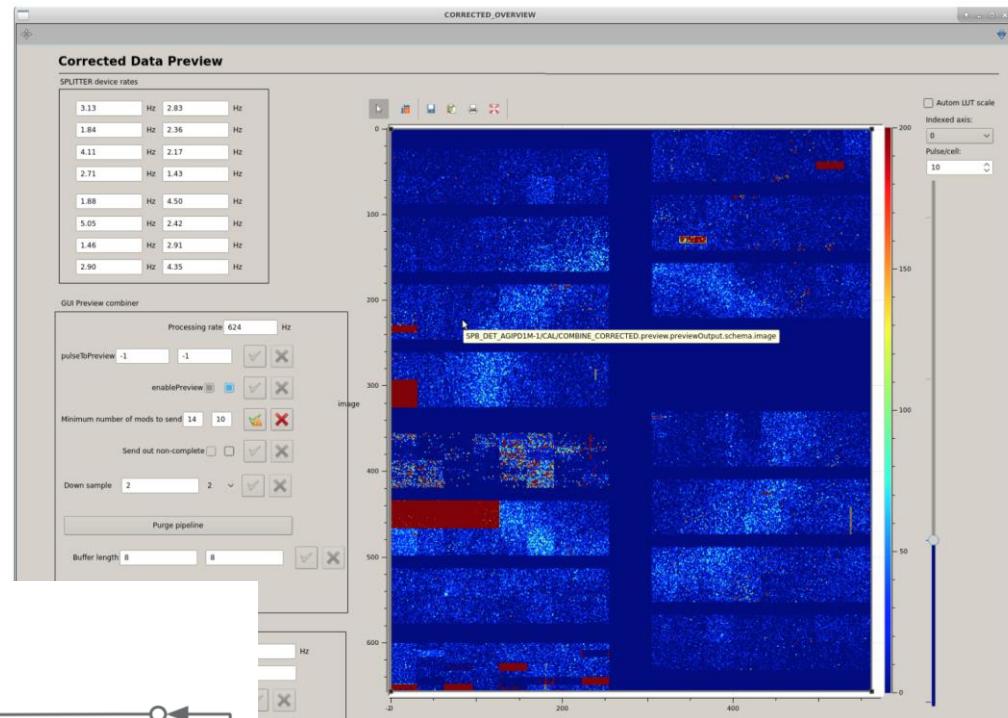
Jupyter notebook

Online Calibration

- Has seen extensive use at both instruments for rapid feedback
- Feeds user-provided online tools via a Karabo-ZMQ adapter device
- GPU algorithms available if data rates require



Corrected online preview for AGIPD

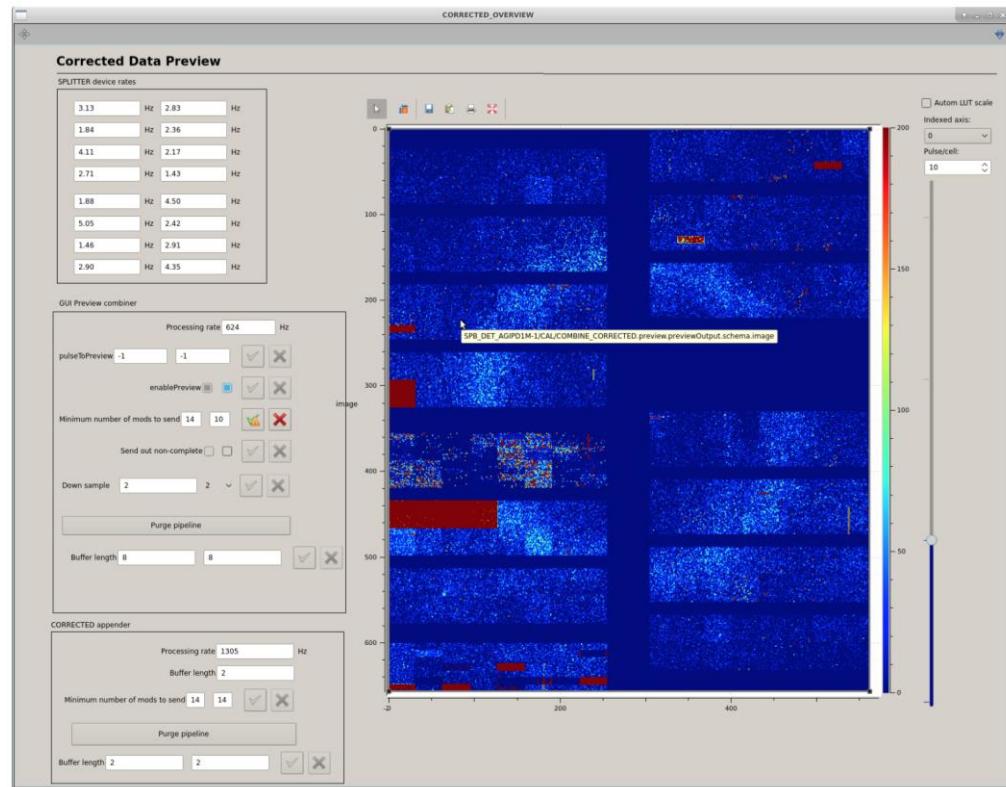


	AGIPD	LPD
Gain evaluation	X	X
Offset correction	X	X
Relative gain correction	X	X
Bad pixels	G/O/N	G/O/N
KRB devices for MPIX	103	69

Online Calibration

- Has seen extensive use at both instruments for rapid feedback
- Feeds user-provided online tools via a Karabo-ZMQ adapter device
- GPU algorithms available if data rates require
- Online processing is module parallel, chunks size always one train
 - “Splitter” devices as entry points to pipelines from → pass every nth train, safe guard DAQ by dropping on slowness
 - “Combiner” device as exit points:
 - ▶ Combine modules into stack: [pulse, module, x y]
 - ▶ Combine modules into image: [pulse, x', y']

Corrected online preview for AGIPD



- Needs to combine 16 data streams @1.6 Gb/s tot.
- Can optionally mask BP
- Configurable number of modules to “wait” for

Online Calibration

- Has seen extensive use at both instruments for rapid feedback
- Feeds user-provided online tools via a Karabo-Bridge device
- GPU algorithms available if data rates require
- Online processing is module parallel, chunks size always one train
 - “Splitter” devices as entry points to pipelines from → pass every nth train, safe guard DAQ by dropping on slowness
 - “Combiner” device as exit points:
 - ▶ Combine modules into stack: [pulse, module, x y]
 - ▶ Combine modules into image: [pulse, x', y']

Corrected online preview for AGIPD

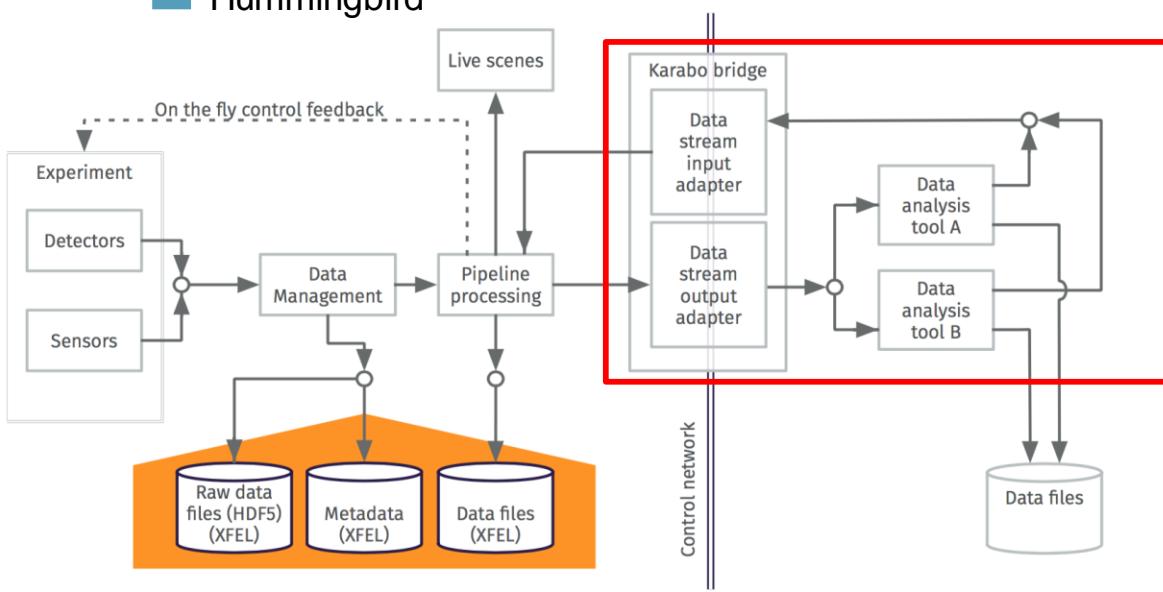
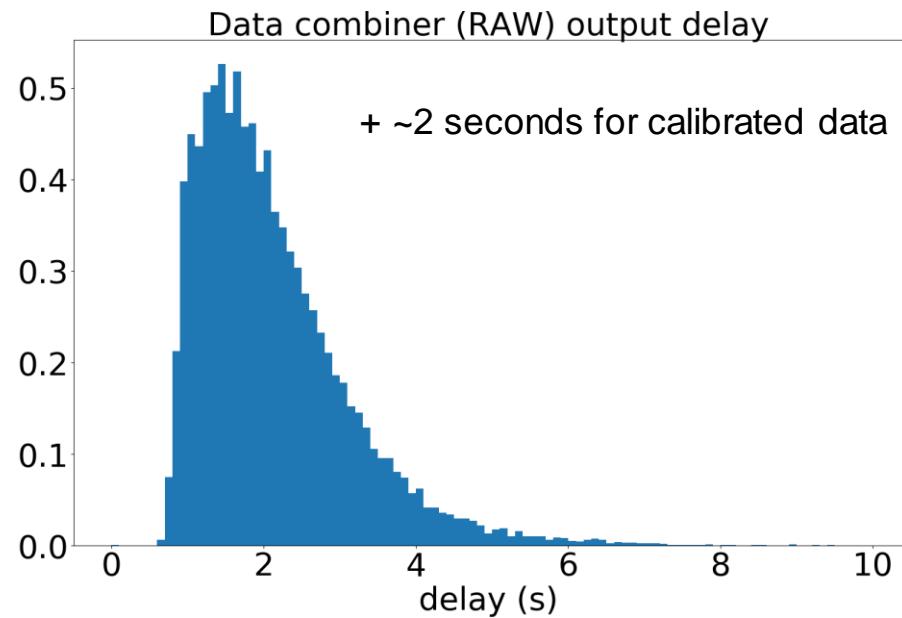


- Needs to combine 16 data streams @1.6 Gb/s tot.
- Can optionally mask BP
- Configurable number of modules to “wait” for

Online Calibration

- Feeds user-provided online tools via a Karabo-Bridge device (T. Michelat)
 - Usually combiner in appender mode
 - Connects to CAS-provided ZMQ bridge
 - 3-5 Hz rate at 64 cells measured
 - 2-4s latency with 64 memory cells

- Applications so far:
 - ONDA
 - CASS
 - Hummingbird



- Latency includes:
 - Data acquisition
 - Data formatting on DAQ
 - Data forwarding to pipelines
 - Data selection at pipeline entry points:
 - ▶ Every nth train
 - ▶ Cells containing FEL pulses
 - Combining of 16 streams from modules
 - Data advertising on ZMQ

Offline Calibration

- Two approaches under investigation:
 - Docker containing Karabo pipeline
 - ▶ Same codebase as online processing
 - ▶ Access to ITDM reader/writers
 - ▶ Unit tests of pyDetLib
 - Python scripts containing correction code
 - ▶ Less overhead
 - ▶ Less opaque
- Common front-end script
- Both run module concurrent on server node and run concurrent on many nodes
- Both feature automated processing whenever data is migrated from online to offline processing via MDC

```
% python calibrate.py --help
usage: calibrate.py [-h] [--input INPUT]
                   [--output OUTPUT]
                   [--base-cal-store BASE_CAL_STORE]
                   [--offset-cal-store OFFSET_CAL_STORE]
                   [--mem-cells MEM_CELLS]
                   [--detector DETECTOR]
                   [--sequences SEQUENCES]
                   [--overwrite] [--no-relgain]
                   [--uuid UUID]
                   [--ff-cal-store FF_CAL_STORE]
                   [--no-ff]

Main entry point for offline calibration

optional arguments:
  -h, --help            show this help message and exit
  --input INPUT
  --output OUTPUT
  --base-cal-store BASE_CAL_STORE
  --offset-cal-store OFFSET_CAL_STORE
  --mem-cells MEM_CELLS
  --detector DETECTOR
  --sequences SEQUENCES
  --overwrite
  --no-relgain
  --uuid UUID
  --ff-cal-store FF_CAL_STORE
  --no-ff
```

Offline Calibration

- Two approaches under investigation:
 - Docker containing Karabo pipeline
 - ▶ Same codebase as online processing
 - ▶ Access to ITDM reader/writers
 - ▶ Unit tests of pyDetLib
 - Python scripts containing correction code
 - ▶ Less overhead
 - ▶ Less opaque
- Common front-end script
- Both run module concurrent on server node and run concurrent on many nodes
- Both feature automated processing whenever data is migrated from online to offline processing via MDC

New	Run in progress			
New	Run Quality			
New	Good (migrate data to Maxwell)			
New	Unclear (migrate data to Maxwell)			
New	Not interesting (data won't be migrated to Maxwell)			
New	Good			
New	Good			
New	Good			
0711	AGIPD Calibration	No Sample	2017-11-09 15:38:29 +0100	Closed
0710	AGIPD Calibration	No Sample	2017-11-09 15:35:14 +0100	Closed
0709	AGIPD Calibration	No Sample	2017-11-09 15:28:00 +0100	Closed
0708	General	No Sample	2017-11-08 19:00:28 +0100	
0707	General	No Sample	2017-11-08 18:36:27 +0100	
0706	General	No Sample	2017-11-08 18:27:50 +0100	

Offline Calibration

- File and directory structure from RAW files is preserved:
 - `./raw/r0012 → ./proc/r0012`
 - All data not touched by corrections is simply copied
- During correction data format may be sanitized and RAW format changes hidden:
 - AGIPD interlaced and AGIPD non-IL have different raw data layouts but same layout in corrected data

	AGIPD	LPD
Gain evaluation	X	X
Offset correction	X	X
Relative gain correction	X	X
Bad pixels	G/O/N	G/O/N
Run time (1000 trains)	~3m	~3m

```
[haufs@max-exfl014]:gpfs/exfel/exp/SPB/201701/p
RAW-R0039-AGIPD00-S00000.h5  RAW-R0039-AGIPD05-
RAW-R0039-AGIPD00-S00001.h5  RAW-R0039-AGIPD05-
RAW-R0039-AGIPD00-S00002.h5  RAW-R0039-AGIPD06-
RAW-R0039-AGIPD00-S00003.h5  RAW-R0039-AGIPD06-
RAW-R0039-AGIPD01-S00000.h5  RAW-R0039-AGIPD06-
RAW-R0039-AGIPD01-S00001.h5  RAW-R0039-AGIPD06-
RAW-R0039-AGIPD01-S00002.h5  RAW-R0039-AGIPD07-
RAW-R0039-AGIPD01-S00003.h5  RAW-R0039-AGIPD07-
RAW-R0039-AGTPD01-S00000.h5  RAW-R0039-AGIPD07-
```



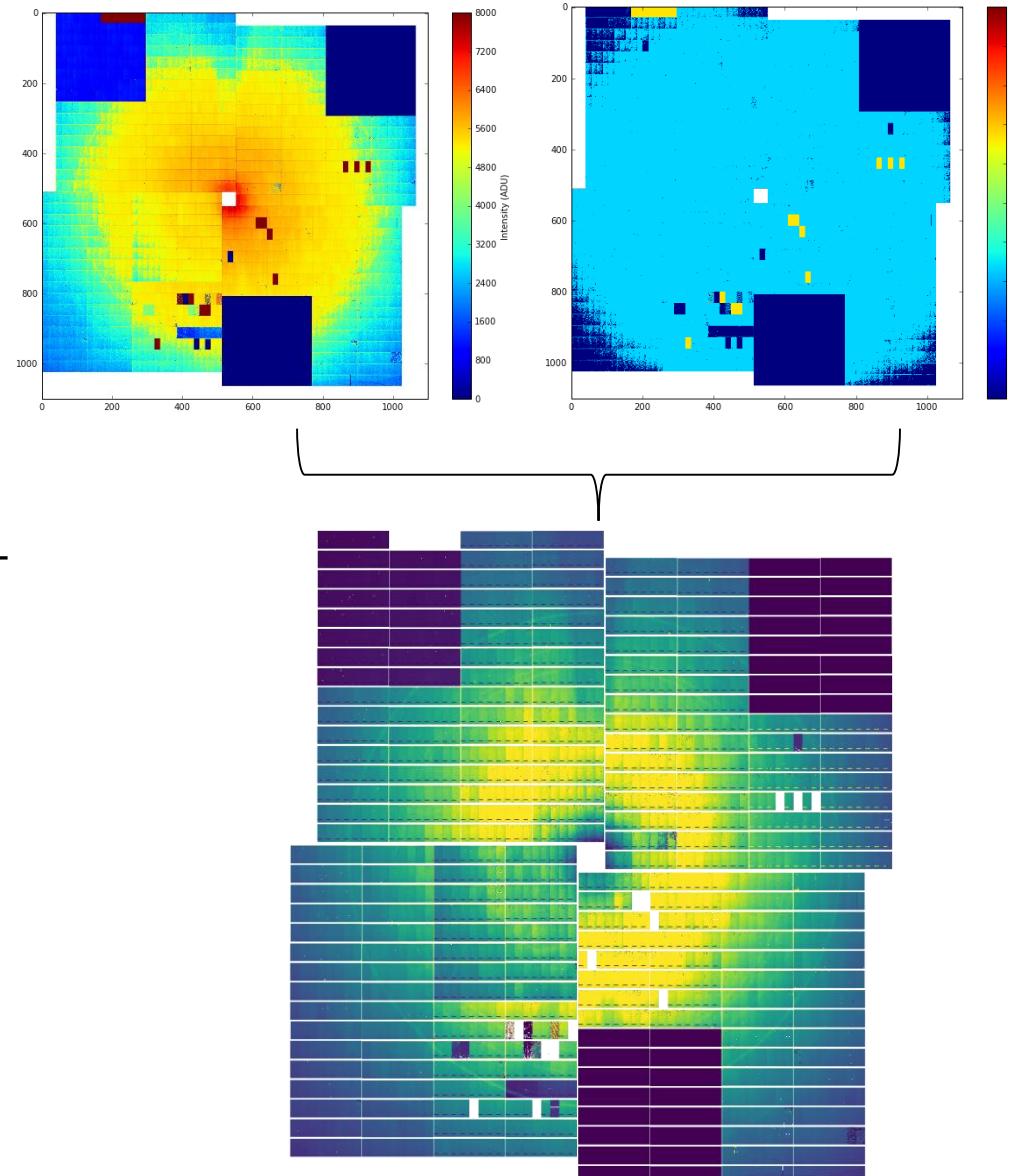
```
[haufs@max-exfl014]:gpfs/exfel/exp/SPB/201701/p
CORR-R0039-AGIPD00-S00000.h5  CORR-R0039-AGIPD00-
CORR-R0039-AGIPD00-S00001.h5  CORR-R0039-AGIPD00-
CORR-R0039-AGIPD00-S00002.h5  CORR-R0039-AGIPD00-
CORR-R0039-AGIPD00-S00003.h5  CORR-R0039-AGIPD00-
CORR-R0039-AGIPD01-S00000.h5  CORR-R0039-AGIPD00-
CORR-R0039-AGIPD01-S00001.h5  CORR-R0039-AGIPD00-
CORR-R0039-AGIPD01-S00002.h5  CORR-R0039-AGIPD00-
CORR-R0039-AGTPD01-S00003.h5  CORR-R0039-AGTPD00-
```

Offline Calibration

- Pending algorithmic improvements:
 - AGIPD gain correction and evaluation
 - ▶ Nonlinear region
 - ▶ Artifacts through offset correction
 - ▶ Bad pixels / outliers
 - LPD
 - ▶ Artifacts at low gain → pin cushion effect
 - ▶ Flat fields needed for CI data with X-ray correlation

- Upcoming technical improvements:
 - Enable usage of remote calibration database
 - Integrate into metadata catalogue

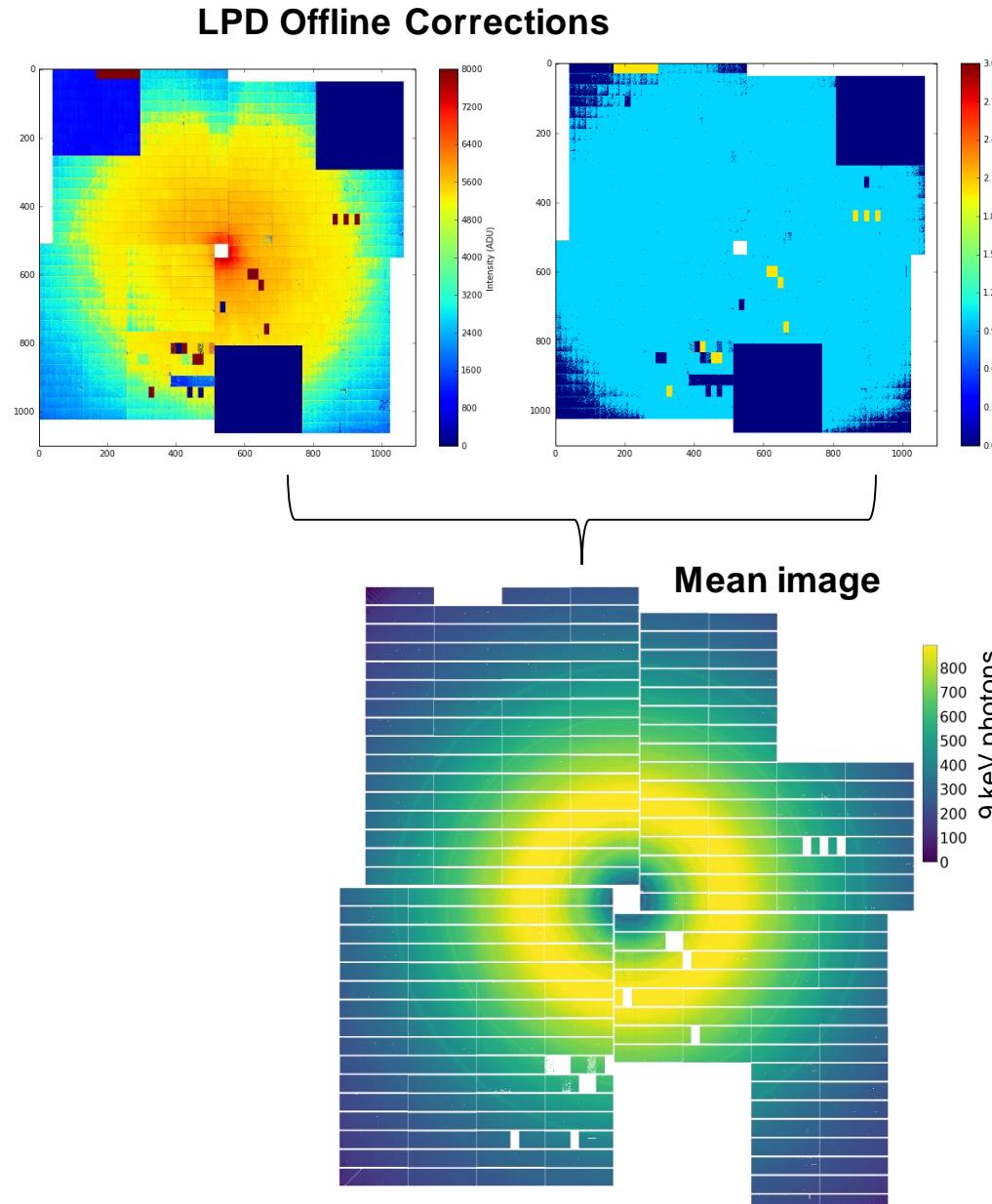
LPD Offline Corrections



Offline Calibration

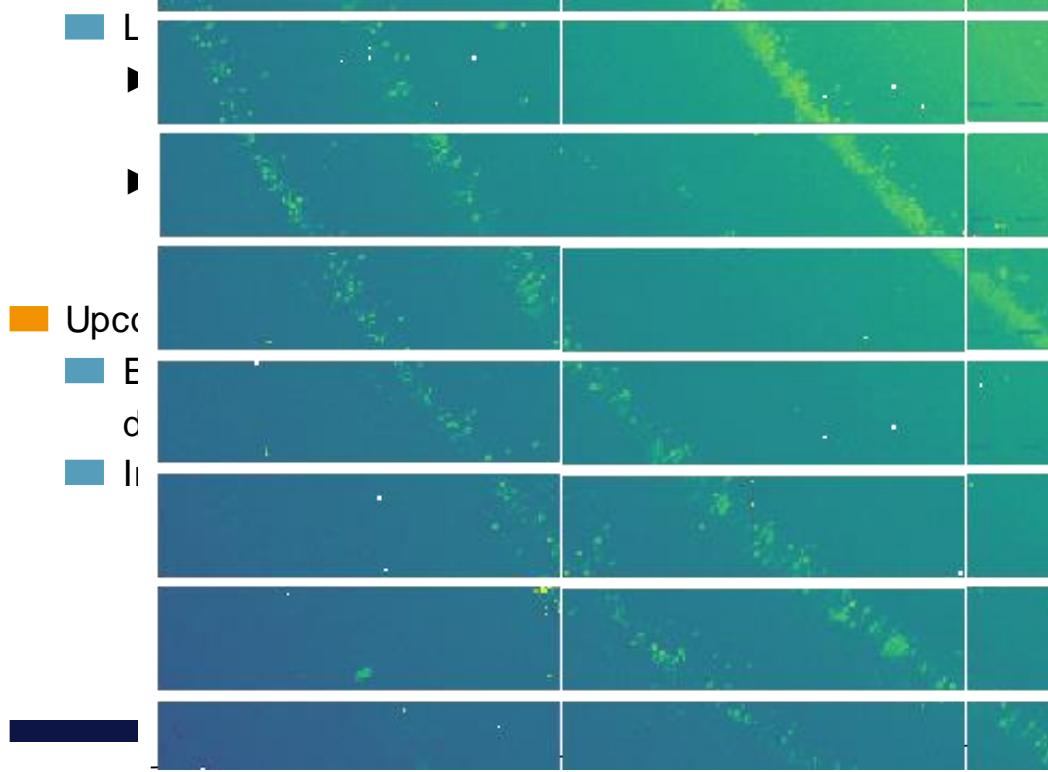
- Pending algorithmic improvements:
 - AGIPD gain correction and evaluation
 - ▶ Nonlinear region
 - ▶ Artifacts through offset correction
 - ▶ Bad pixels / outliers
 - LPD
 - ▶ Artifacts at low gain → pin cushion effect
 - ▶ **Flat fields needed for CI data with X-ray correlation**

- Upcoming technical improvements:
 - Enable usage of remote calibration database
 - Integrate into metadata catalogue

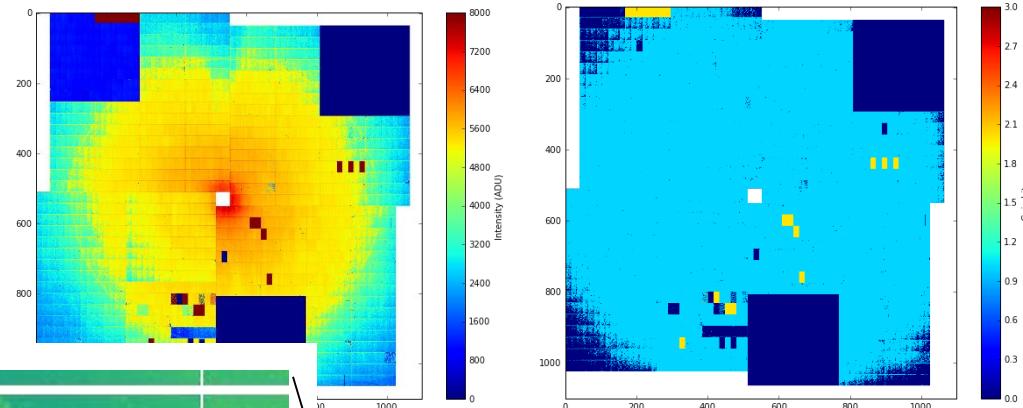


Offline Calibration

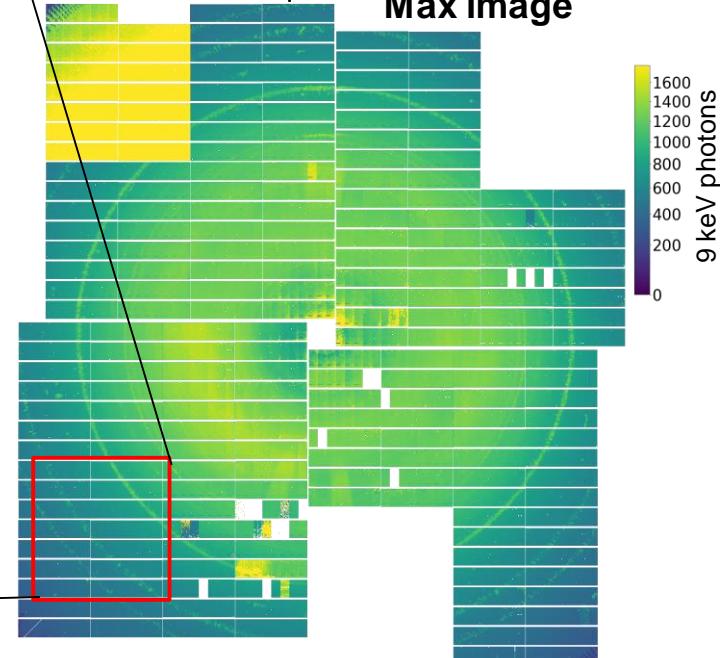
- Pending algorithmic improvements:
 - AGIPD gain correction and evaluation
 - ▶ Nonlinear region
 - ▶ Artifacts through offset correction
 - ▶ Dark current subtraction



LPD Offline Corrections



Max image

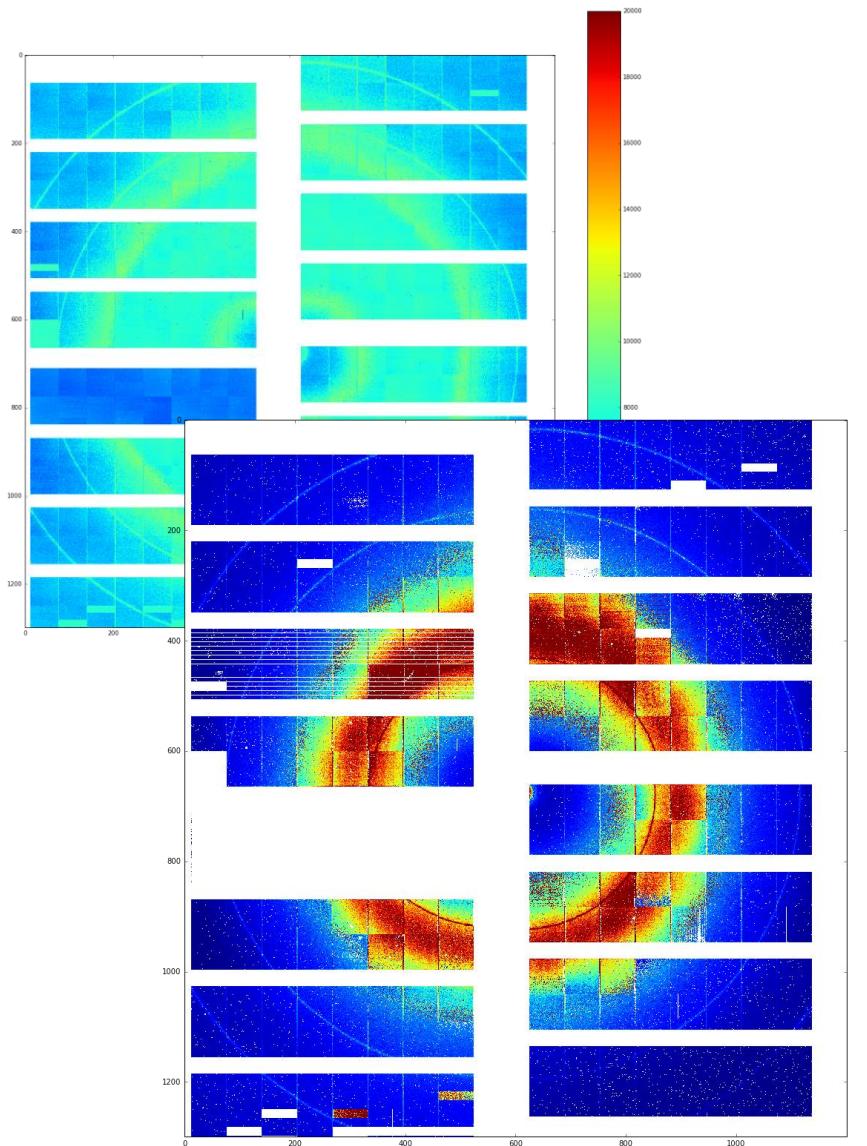


Offline Calibration

- Pending algorithmic improvements:
 - AGIPD gain correction and evaluation
 - ▶ Nonlinear region
 - ▶ Artifacts through offset correction
 - ▶ Bad pixels / outliers
 - LPD
 - ▶ Artifacts at low gain → pin cushion effect
 - ▶ Flat fields needed for CI data with X-ray correlation

- Upcoming technical improvements:
 - Enable usage of remote calibration database
 - Integrate into metadata catalogue

AGIPD Offline Corrections

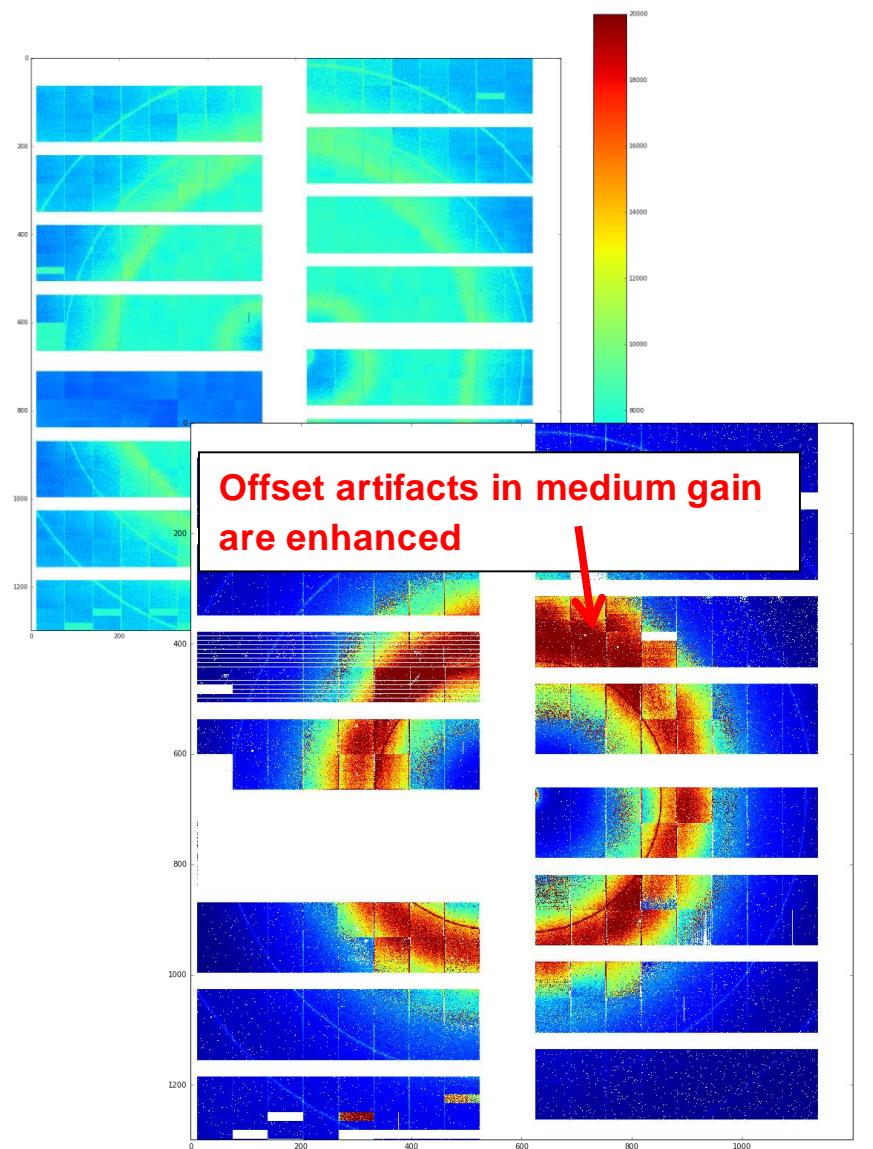


Offline Calibration

- Pending algorithmic improvements:
 - AGIPD gain correction and evaluation
 - ▶ Nonlinear region
 - ▶ Artifacts through offset correction
 - ▶ Bad pixels / outliers
 - LPD
 - ▶ Artifacts at low gain → pin cushion effect
 - ▶ Flat fields needed for CI data with X-ray correlation

- Upcoming technical improvements:
 - Enable usage of remote calibration database
 - Integrate into metadata catalogue

AGIPD Offline Corrections

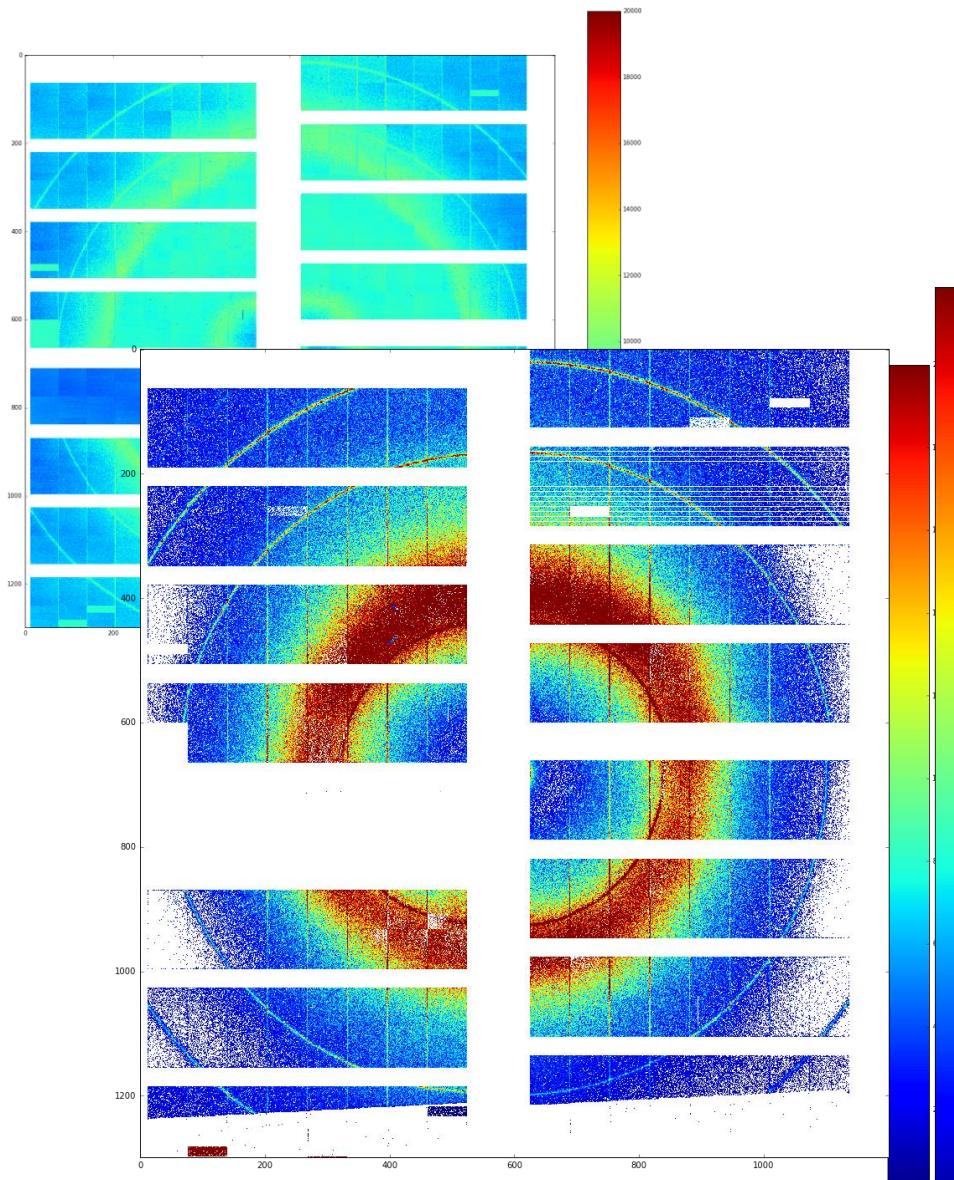


Offline Calibration

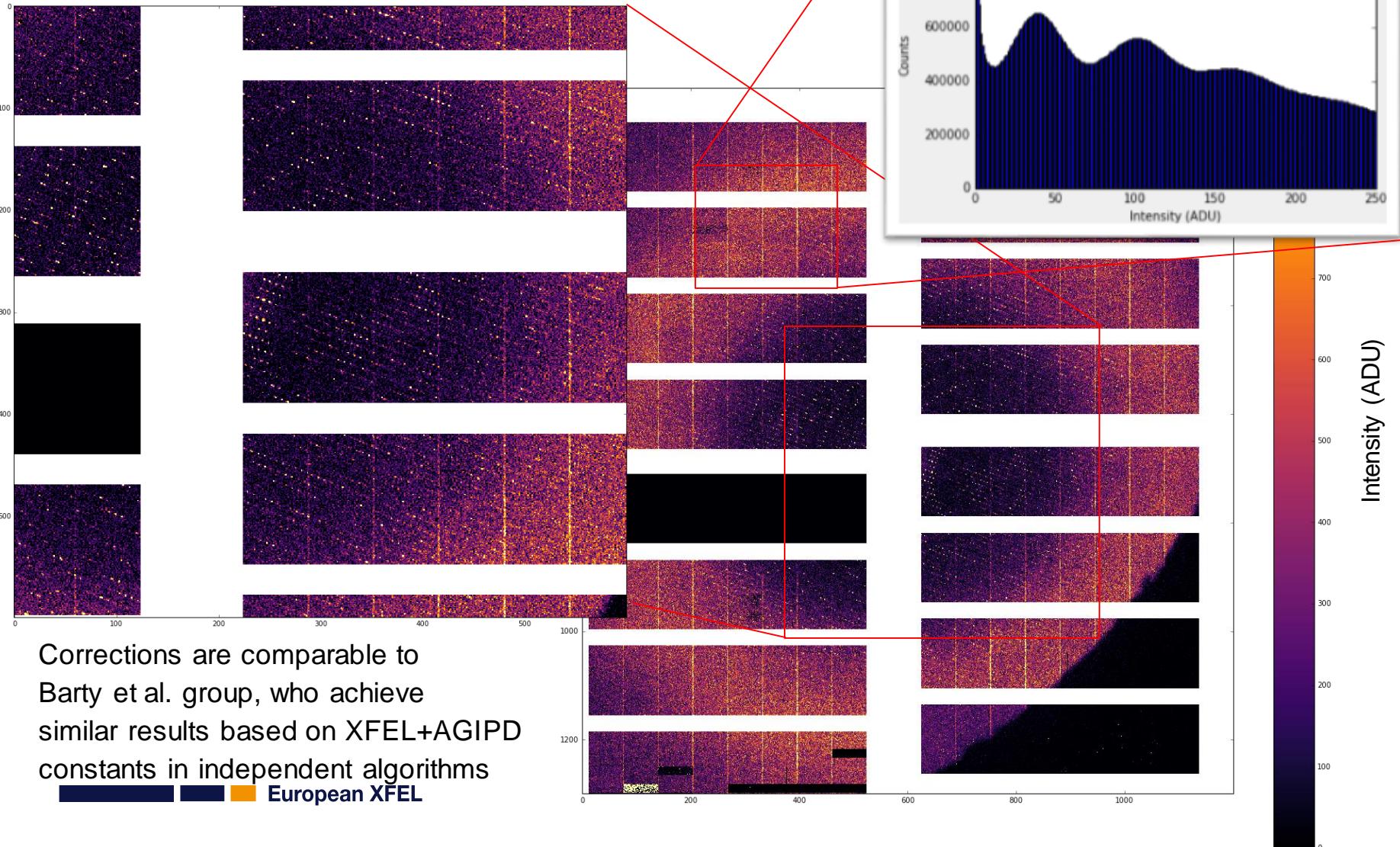
- Pending algorithmic improvements:
 - AGIPD gain correction and evaluation
 - ▶ Nonlinear region
 - ▶ Artifacts through offset correction
 - ▶ Bad pixels / outliers
 - LPD
 - ▶ Artifacts at low gain → pin cushion effect
 - ▶ Flat fields needed for CI data with X-ray correlation

- Upcoming technical improvements:
 - Enable usage of remote calibration database
 - Integrate into metadata catalogue

AGIPD Offline Corrections



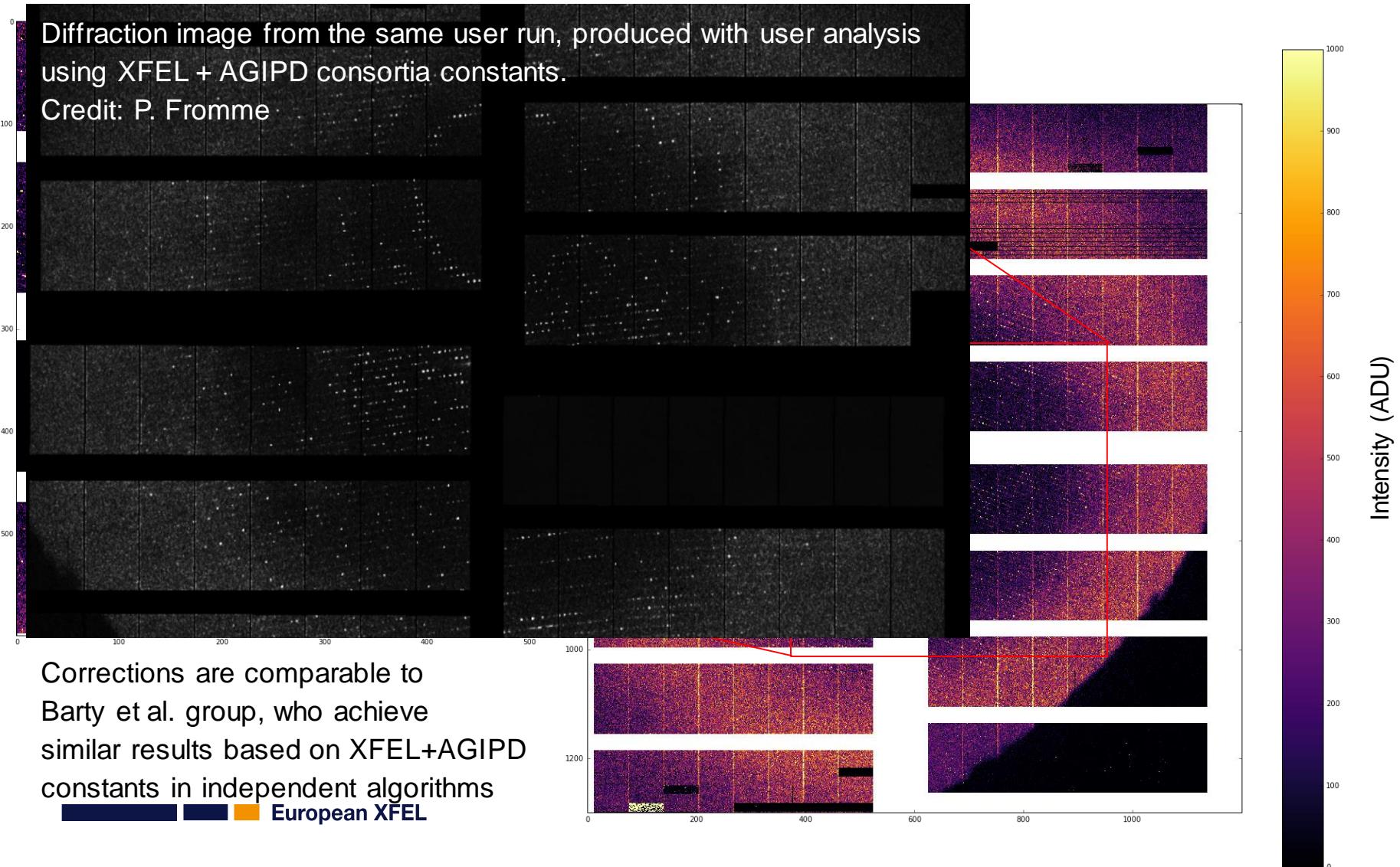
Corrected AGIPD Data – Verification



Corrected AGIPD Data – Verification

Diffraction image from the same user run, produced with user analysis using XFEL + AGIPD consortia constants.

Credit: P. Fromme



Characterization Processing

- Frequently use Jupyter notebooks

- For AGIPD:

- Dark image analysis → offset, noise, bad pixels, gain thresholds
- Pulse capacitor data → medium/high gain relation, bad pixels, gain thresholds
- Current source data → medium/low gain relation, bad pixels, gain thresholds
- Flat fields: → X-ray/charge injection relation, bad pixels

Frequent updates,
5min processing

More static,
Each about 2-4 hours
processing each

- For LPD

- Dark image analysis → offset, noise, bad pixels
- Charge injection data → gain stage relations, bad pixels
- Flat fields: → X-ray/charge injection relation, bad pixels

Frequent updates,
5min processing

More static,
Each about 2-4 hours
processing each

Characterization Processing

- Frequently use Jupyter notebooks

- For AGIPD:

- Dark image analysis → offset, noise, bad pixels, gain thresholds
- Pulse capacitor data → medium/high gain relation, bad pixels, gain thresholds
- Current source data → medium/low gain relation, bad pixels, gain thresholds
- Flat fields: → X-ray/charge injection relation, bad pixels

Frequent updates,
5min processing

More static,
Each about 2-4 hours
processing each

- For LPD

- Dark image analysis → offset, noise, bad pixels
- **Charge injection data → gain stage relations, bad pixels**
- Flat fields: → X-ray/charge injection relation, bad pixels

]

Data is produced via Karabo processing pipelines storing only pixels with charge injected: 1.75 TB/d, 10GB/s peak

Characterization Processing

■ Frequently use Jupyter notebooks

■ For AGIPD:

- Dark image analysis → offset, noise, bad pixels, gain thresholds
- Pulse capacitor data → medium/high gain relation, bad pixels, gain thresholds
- Current source data → medium/low gain relation, bad pixels, gain thresholds
- Flat fields: → X-ray/charge injection relation, bad pixels

■ For LPD

- Dark image analysis → offset, noise, bad pixels
- Charge injection data → gain stage relations, bad pixels
- Flat fields: → X-ray/charge injection relation, bad pixels

Characterize AGIPD Pulse Capacitor Data

The following code characterizes AGIPD gain via data take with the pulse capacitor so scanning through the high and medium gains of AGIPD, by subsequently increasing a on-ASIC capicitor, thus increasing the charge a pixel sees in a given integration time.

Because induced charge does not originate from X-rays on the sensor, the gains evaluated with gains deduced from X-ray data.

PCS data is organized into multiple runs, as the on-ASIC current source cannot supply charge at the same time. Hence, only certain pixel rows will have seen charge for a given integration time, need to be combined into single module images again.

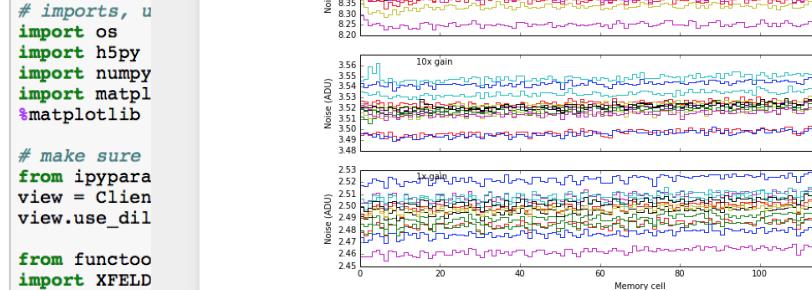
We then use a K-means clustering algorithm to identify components in the resulting pedestal images. We then use three general regions:

- a high gain slope
- a transition region, where gain switching occurs
- a medium gain slope.

The same regions are present in the gain-bit data and are used to deduce the switching regions.

The resulting slopes can then be used to determine the regions in which the data is taken.

```
[1]: # imports, u
      import os
      import h5py
      import numpy
      import matplotlib
      %matplotlib
```

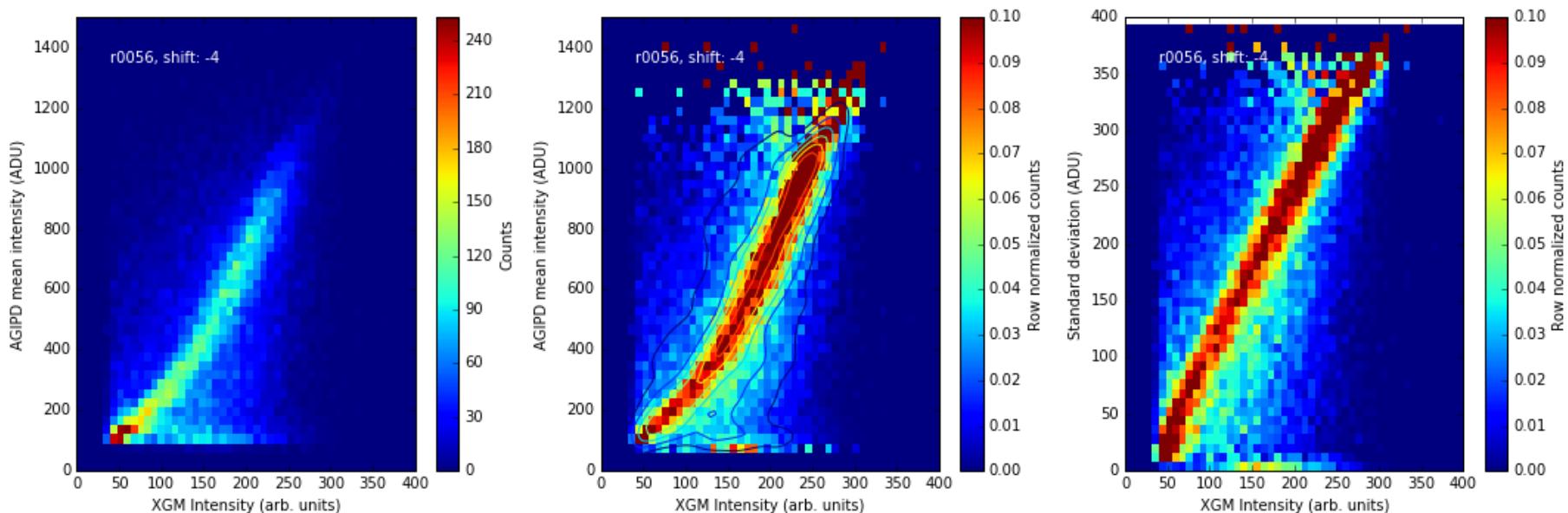


```
In [12]: # save everything to file.
for cap in capacitor_settings:
    runs = [v for k, v in gain_runs.items() if cap in k]
    file = "{}/lpd_offset_store_{}.h5".format(out_folder, cap)
    store_file = h5py.File(file, "w")
    for qm in offset_g[cap].keys():
        store_file["{}/Offset/0/data".format(qm)] = offset_g[qm]
        store_file["{}/Noise/0/data".format(qm)] = noise_g[qm]
        store_file["{}/BadPixelsDark/0/data".format(qm)] = badpix_g[qm]
    store_file.close()
```

```
In [13]: def show_hists(gain_to_preview, cap, ranges):
    res = OrderedDict()
    for i in range(16):
        qm = "Q{}M{}".format(i//4+1, i%4+1)
        try:
            res[qm] = OrderedDict()
            res[qm]['Offset'] = offset_g[cap][qm]
            res[qm]['Noise'] = noise_g[cap][qm]
            res[qm]['BadPixels'] = copy.copy(badpix_g[qm])
            res[qm]['BadPixels'] = np.concatenate(res[qm]['BadPixels'])
```

Characterization Processing – Data Correlation with XGM

AGIPD: module mean and std dev. vs XGM signal

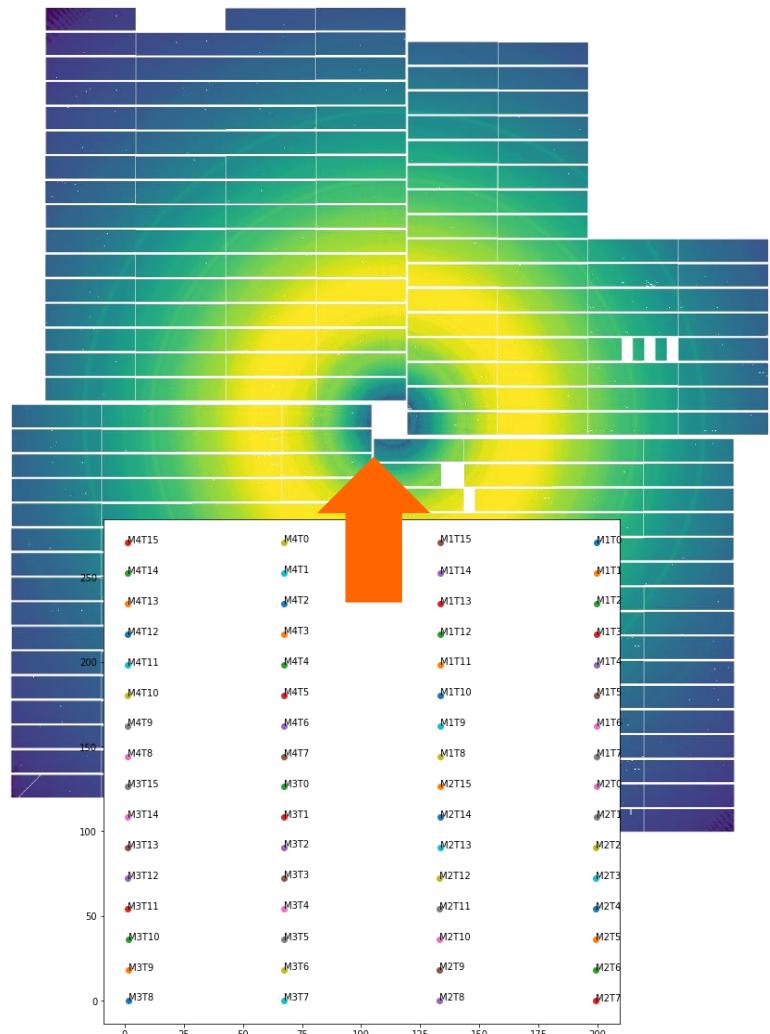


- Requires correlation between different data sources on a per-pulse level → relatively straight forward using indexing meta-data

Alignment Data

- Goal: Facility-side provide initial module positions on segmented detectors → possible refine in situ if needed
- In verification with FXE for LPD for positions up to quadrant level determined using photographic metrology data (procedure and analysis by T. Rüter)
- AGIPD to be produced
- Data store in HDF5 file matching the hierarchical layout of the detector, will be made available through calibration database in the future.
- Python code provided to get position of a given module in terms of top left corner pixel.
- Within module lithography assumed as “perfect”

Corrected LPD (5 train mean, modules positioned)

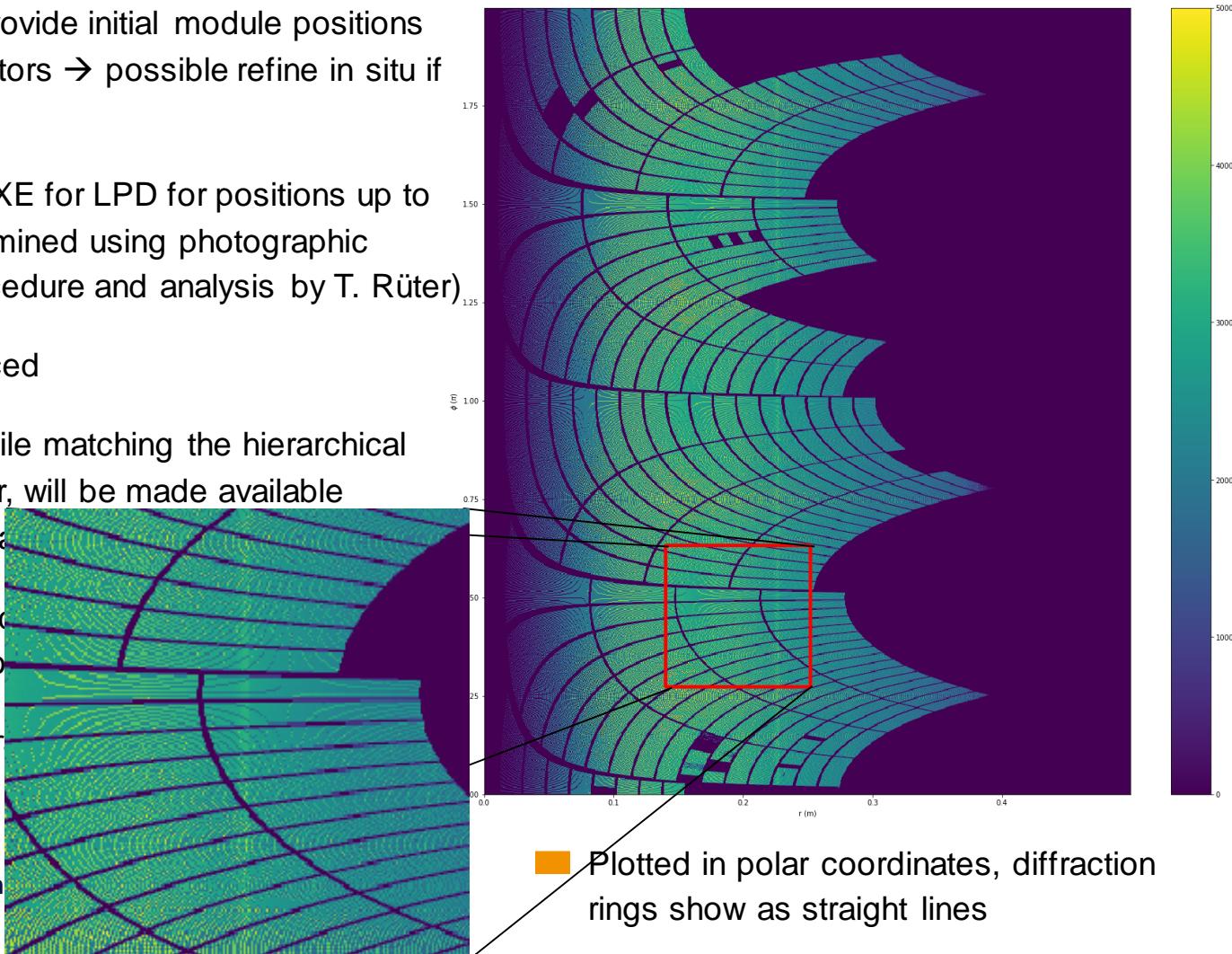


Alignment Data

- Goal: Facility-side provide initial module positions on segmented detectors → possible refine in situ if needed
- In verification with FXE for LPD for positions up to quadrant level determined using photographic metrology data (procedure and analysis by T. Rüter)
- AGIPD to be produced
- Data store in HDF5 file matching the hierarchical layout of the detector, will be made available through calibration data
- Python code provided to move module in terms of to
- Within module lithograph

■ European

Corrected LPD (5 train mean, modules positioned)

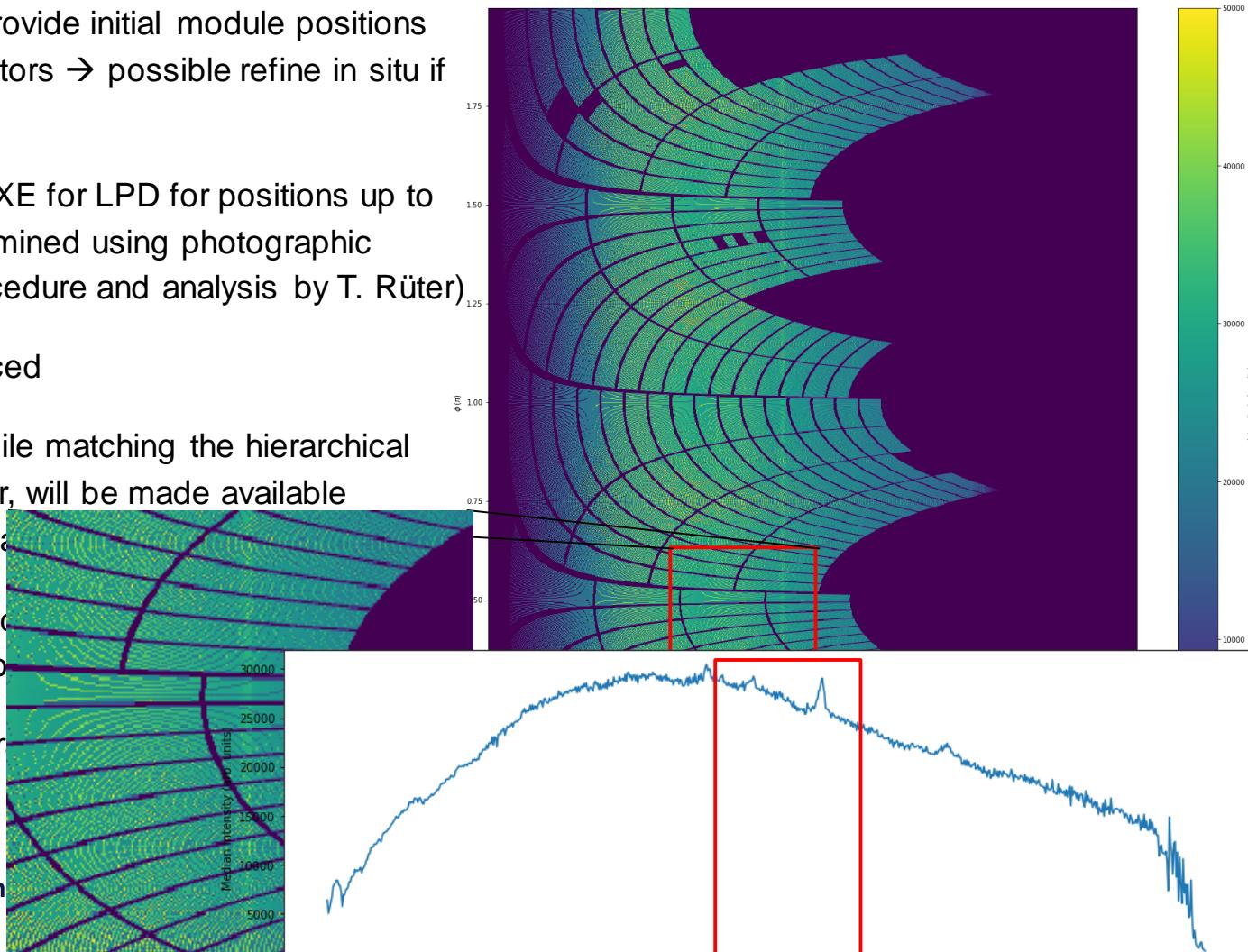


Alignment Data

- Goal: Facility-side provide initial module positions on segmented detectors → possible refine in situ if needed
- In verification with FXE for LPD for positions up to quadrant level determined using photographic metrology data (procedure and analysis by T. Rüter)
- AGIPD to be produced
- Data store in HDF5 file matching the hierarchical layout of the detector, will be made available through calibration data
- Python code provided to align each module in terms of top-left corner
- Within module lithography alignment

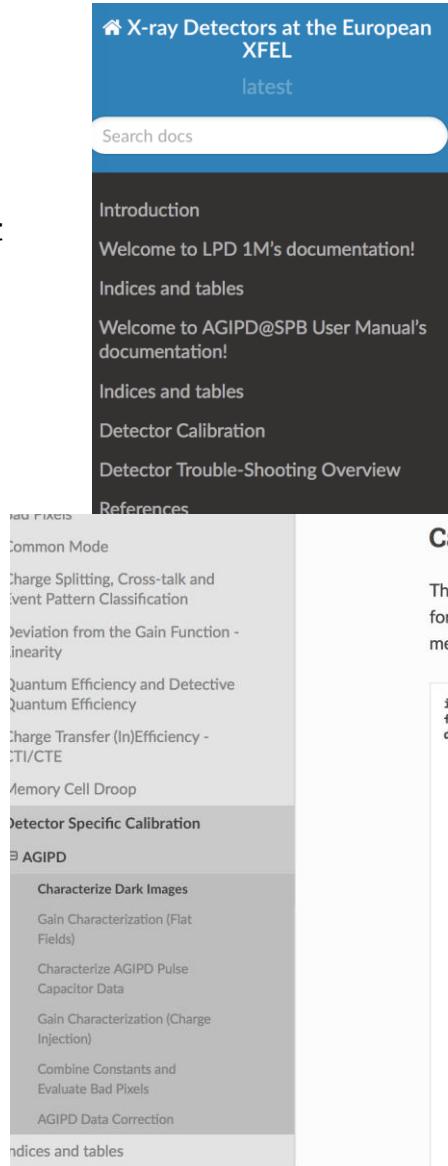
■ European ■ American

Corrected LPD (5 train mean, modules positioned)



Documentation

- Documentation is on readthedocs:
<https://in.xfel.eu/readthedocs/docs/detector-documentation/en/latest/index.html>
- General detector documentation
- General calibration documentation
- Detector specific calibration documentation
 - Also documents how constants are derived



Docs » Welcome to X-ray Detectors at the

Welcome to X-ray Detector documentation!

Contents:

- [Introduction](#)
- [Welcome to LPD 1M's documentation!](#)
 - [Introduction](#)
 - [Hardware Setup](#)
 - [Software Setup](#)
 - [Configuration and Operation](#)

Calculate Offsets, Noise and Thresholds

action:

The calculation is performed per-pixel and per-memory-cell. Offset for a set of dark data taken at a given gain, noise the standard deviation of the medians of the gain array.

• [Manual](#)

```
import copy
from functools import partial
def characterize_module(cells, inp):
    import numpy as np
    import copy
    import h5py

    filename, filename_out, channel = inp

    infile = h5py.File(filename, "r", driver="core")
    im = np.array(infile["/INSTRUMENT/SPB_DET_AGIPD1M-1/DET/{}CH0:xtd".format(channel)])
    infile.close()

    ga = im[1::2, 0, ...]
    im = im[0::2, 0, ...].astype(np.float32)

    im = np.rollaxis(im, 2)
    im = np.rollaxis(im, 2, 1)

    ga = np.rollaxis(ga, 2)
    ga = np.rollaxis(ga, 2, 1)

    offset = np.zeros((im.shape[0], im.shape[1], cells//2))
    gains = np.zeros((im.shape[0], im.shape[1], cells//2))
    noise = np.zeros((im.shape[0], im.shape[1], cells//2))

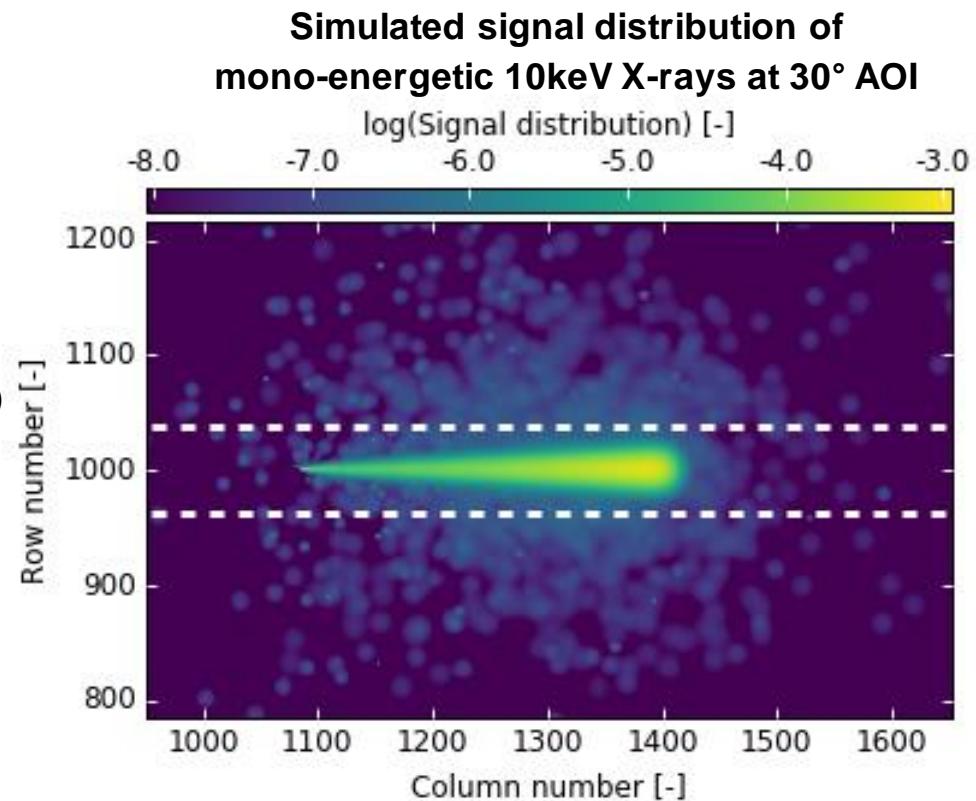
    for cc in range(cells//2):
        offset[...,cc] = np.median(im[..., cc::cells//2], axis=2)
        noise[...,cc] = np.std(im[..., cc::cells//2], axis=2)
        gains[...,cc] = np.median(ga[..., cc::cells//2], axis=2)
```

or at SPB

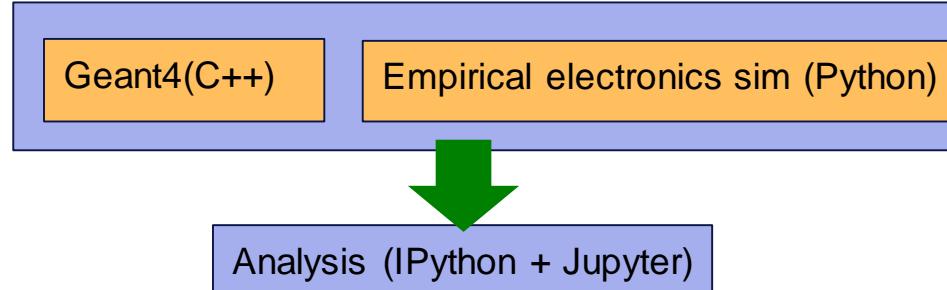
tor at SPB

Simulation Activities: Angular Resolution Studies

- Work by T. Rüter
- Simulation framework now allows to study event mis-location for large parameter space:
 - Energy (6 keV – 25 keV, 1 keV steps)
 - Pixel size (> 1um, min. bin size)
 - Angle of incidence (0° - 50°, 2.5° steps)
- Validation experiment planned jointly with Uni Siegen for 2018
- Work has been presented at IEEE NSS&MIC 2017, publication of results as peer-reviewed paper planned
- Mid-term goals: study implication on representative XFEL experiments and evaluate in combination with accuracy of metrology measurements (p.o.c for LPD exists)



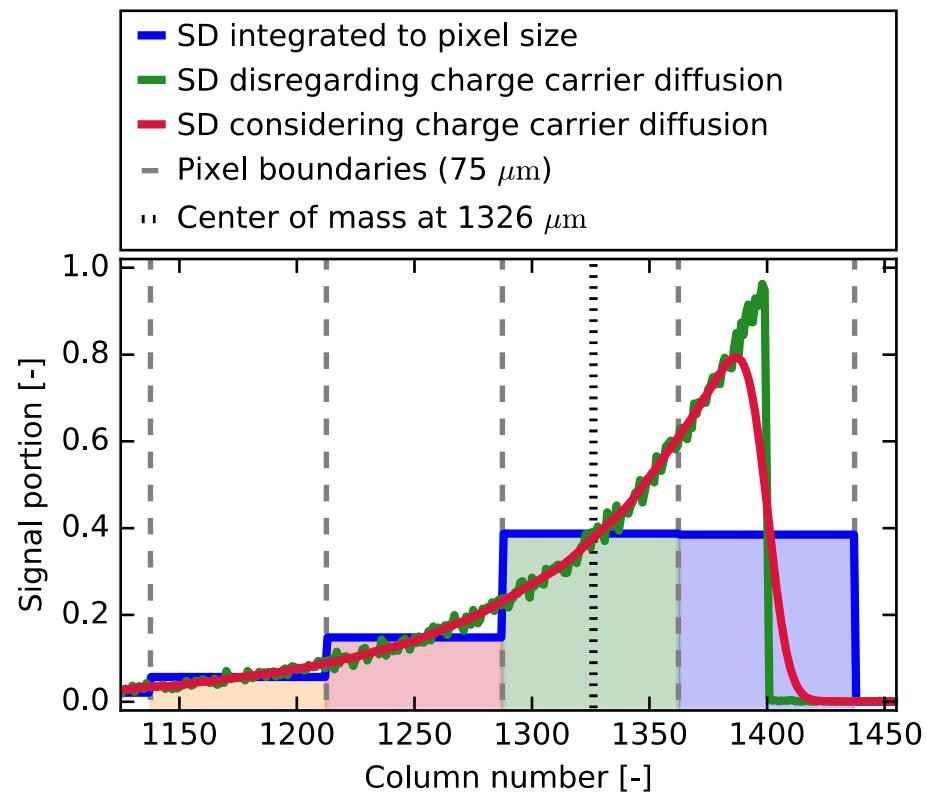
Karabo



Simulation Activities: Angular Resolution Studies

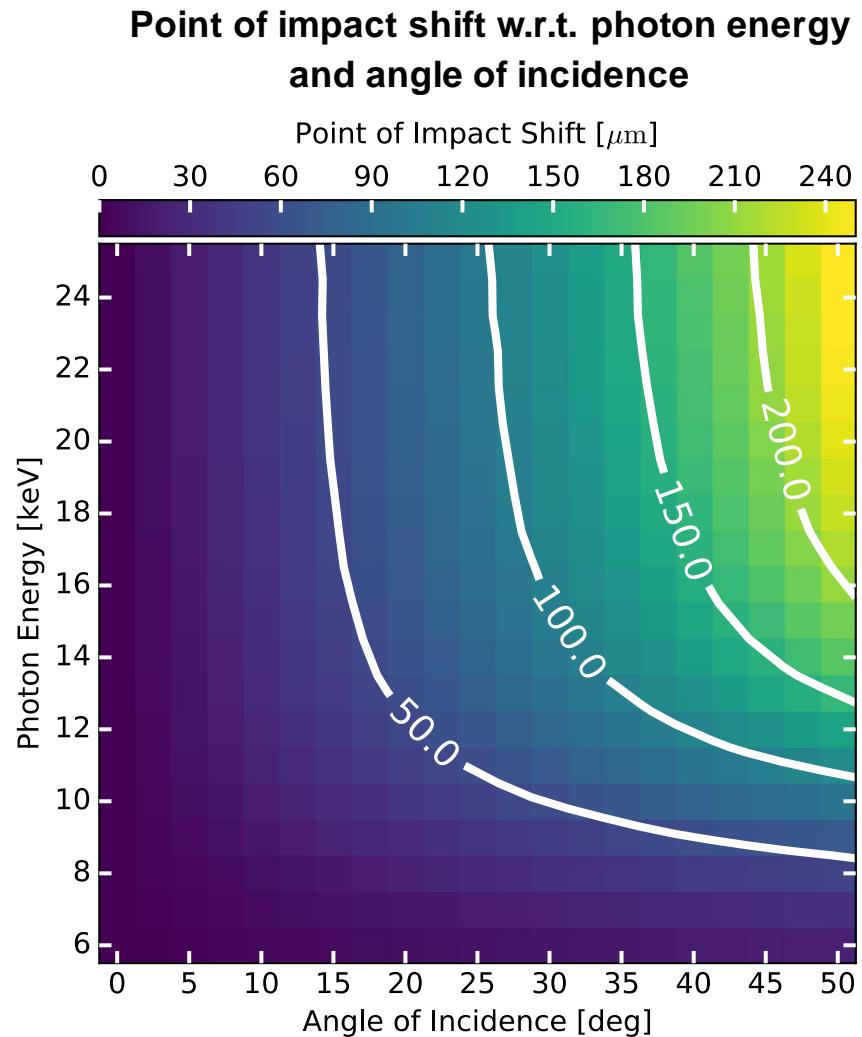
- Work by T. Rüter
- Simulation framework now allows to study event mis-location for large parameter space:
 - Energy (6 keV – 25 keV, 1 keV steps)
 - Pixel size (> 1um, min. bin size)
 - Angle of incidence (0° - 50°, 2.5° steps)
- Validation experiment planned jointly with Uni Siegen for 2018
- Work has been presented at IEEE NSS&MIC 2017, publication of results as peer-reviewed paper planned
- Mid-term goals: study implication on representative XFEL experiments and evaluate in combination with accuracy of metrology measurements (p.o.c for LPD exists)

1-D charge distribution in sensor and effect on 75um detector



Simulation Activities: Angular Resolution Studies

- Work by T. Rüter
- Simulation framework now allows to study event mis-location for large parameter space:
 - Energy (6 keV – 25 keV, 1 keV steps)
 - Pixel size (> 1um, min. bin size)
 - Angle of incidence (0° - 50°, 2.5° steps)
- Validation experiment planned jointly with Uni Siegen for 2018
- Work has been presented at IEEE NSS&MIC 2017, publication of results as peer-reviewed paper planned
- Mid-term goals: study implication on representative XFEL experiments and evaluate in combination with accuracy of metrology measurements (p.o.c for LPD exists)



Summary

- Online feedback (raw & calibrated) and offline corrections are available to instruments and users during beam times
 - Continuous improvements since first FEL light on AGIPD and LPD in August
 - Good and frequent feedback from instruments and users
 - By now: low latency with max. few seconds delay
- Online data can be passed to user tools via CAS-provided Karabo bridge
- Offline corrections still being optimized for both detectors
 - More characterization data is needed → dedicated commissioning time
 - Algorithms also still being optimized as result of further characterization

Outlook

■ December/January shutdown

- Remote calibration database for online and offline
- Triggering and configuring offline calibration from metadata catalogue
- Optimization of characterization for both detectors
- (GPUs on online calibration)
- Initial DSSC integration

■ Longer term

- Enable full GPU support on offline
- Automated performance tracking of selected parameters and signals of interest
- DSSC integration
- User pluggable components in offline calibration allowing for pre-selection
- Calibration data release policy & quality assurance

■ Technical and performance papers to cite are under preparation