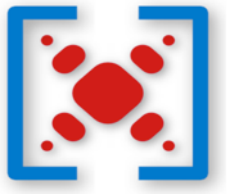
Computer exercises

Introduction to serial femtosecond crystallography data analysis with CrystFEL

Nadia Zatsepin

Arizona State University



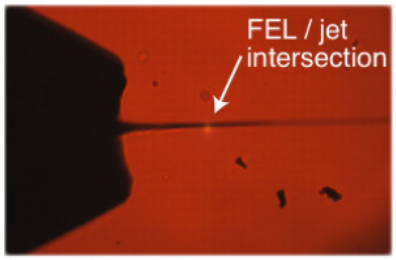
Serial nanocrystallography data analysis exercises outline

- SFX data analysis overview
- Raw data reduction: Cheetah
- Overview of CrystFEL: indexing, merging, scaling
- CrystFEL basic usage with real XFEL data
- Merging, point groups and indexing ambiguities
- Importing data into CCP4
- How you can contribute to CrystFEL
- Future developments and discussion session

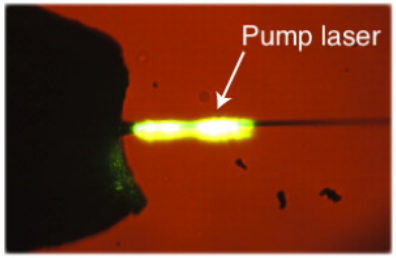
Pump probe SFX setup 1

single excitation of Photosystem I – Ferredoxin cocrystals

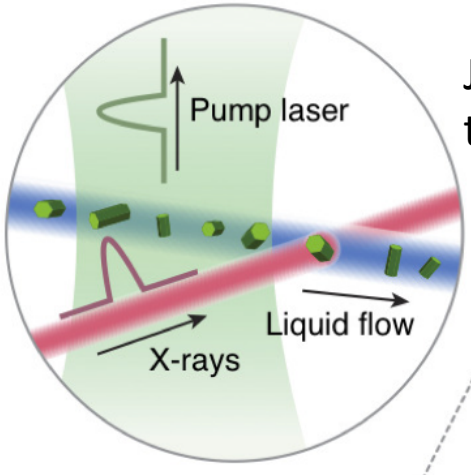
Crystals are in liquid jet, not on a goniometer



Pump laser off



Pump laser on



John Spence will talk about this tomorrow

Setup at AMO, LCLS, in 2010

Aquila A. et al. Opt Express 2012, 20:2706-2716. Time-resolved protein nanocrystallography using an X-ray free-electron laser.

XFEL beam characteristics

- Very high intensity: 10^{12} – 10^{13} photons/pulse (average)
- Ultrashort X-ray pulses: 10 – 200 fs
- Pulse repetition rate: 120 Hz at LCLS, **27 kHz** at European XFEL (planned)
- Tunable X-ray wavelength: 1.5 – 6.2 Å (2-9keV) at LCLS
- Full transverse coherence
- SASE bandwidth: ~ 0.1 %
- Seeded beam bandwidth: 10^{-4}
- Small bandwidth (it's a laser!) so no Laue crystallography for time-resolved
- Current smallest LCLS beam size: 0.1 µm diameter

Synchrotron beam characteristics

- Photon flux: 10^{11} – 10^{13} /s
- X-ray pulses down to <100 ps, but generally longer
- Tunable X-ray wavelength:

  generally 0.5 - 2.0 Å or 5-15 keV

- Limited transverse coherence
- Tunable bandwidth (from pink beam, 2-3% bw, to 0.001% bw with a Si 311 mono)
- Can focus X-rays to ~ 40nm at 15 keV; even smaller beams have been reported

XFEL

serial femtosecond crystallography

- Nano/microcrystals: typically 0.1 – 20 µm
- outrun radiation damage (diffract before destroy)
- measure snapshots in time (static sample) – one pattern per crystal
- Room temperature
- crystals are in liquid or lipidic jet, not on a goniometer
- dynamic studies, e.g. pump probe
- Need at least 10,000 indexed patterns (individual crystals) for one dataset
- High sample consumption (low hit rate)
- LCP jet for low sample consumption, e.g. of GPCRs

Synchrotron crystallography

- Need “large”, high-quality crystal
  - Typically 10 – 1000 µm
- Crystals are usually frozen, but radiation damage is still a constraint
- Each exposure: 3 - 20 s
- up to 1000 frames per data set
- 1-10 crystals for a dataset
- Can do Laue or convergent beam crystallography

XFEL

serial femtosecond crystallography

- Nano/microcrystals
  - typically 0.1 – 20 µm
- outrun radiation damage (diffract before destroy)
- measure snap shots in time (static sample) – one pattern per crystal
- Room temperature
- crystals are in liquid or lipidic jet
- dynamic studies, e.g. pump probe
- Need at least 10,000 indexed patterns (individual crystals) for one dataset
- High sample consumption (low hit rate)
- LCP jet for low sample consumption, e.g. of GPCRs

implications

- **Easier to grow** small crystals (usually)
- nanocrystals could be of **higher quality** than large crystals
  - There are enough photons to see high resolution diffraction from tiny crystals
- Bragg spots are broadened by "**shape transform"** of nanocrystal
  - This can be exploited in a good way!
- Nearly all Bragg spots are **partially** recorded (no oscillation)
- **Undamaged** structures
  - Higher resolution, even from very radiation-sensitive samples
- Naturally occurring conformations
  - Some xtals are damaged by freezing
  - Avoid ice formation
- **Time resolved** studies in **biologically relevant** environment

Main point about serial femtosecond crystallography

- Because the beam destroys the sample, a continuously refreshed supply of nanocrystals is needed.
- Every X-ray pulse hits a single crystal and the resulting diffraction pattern is then read out before the next shot arrives
- This is repeated 120 times/second
- Many **thousands** of patterns are acquired, each containing partial (snap shot) reflections

- **Final reflection list is from thousands of different crystals!**
- Hits occur by chance, due to choice of sample concentration. Many X-ray pulses do not hit crystals. A synchronized system is possible.

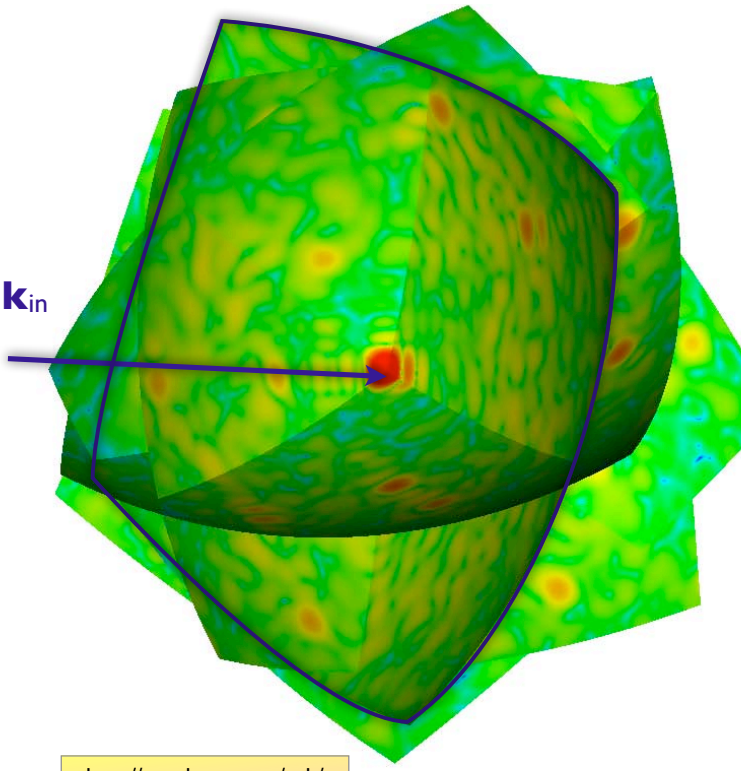
Merging SFX snapshots: Monte Carlo integration

Every pattern = new crystal; single 2D Ewald sphere (or thin shell)

When we merge the data in 3D, we integrate over thousands of different nano/microcrystal sizes, shapes, orientations and quality, as well as X-ray pulse fluctuations, such as intensity, energy and bandwidth.

Given enough patterns, it works!

Kirian et al., Opt. Exp. **18** (2010) 5713.



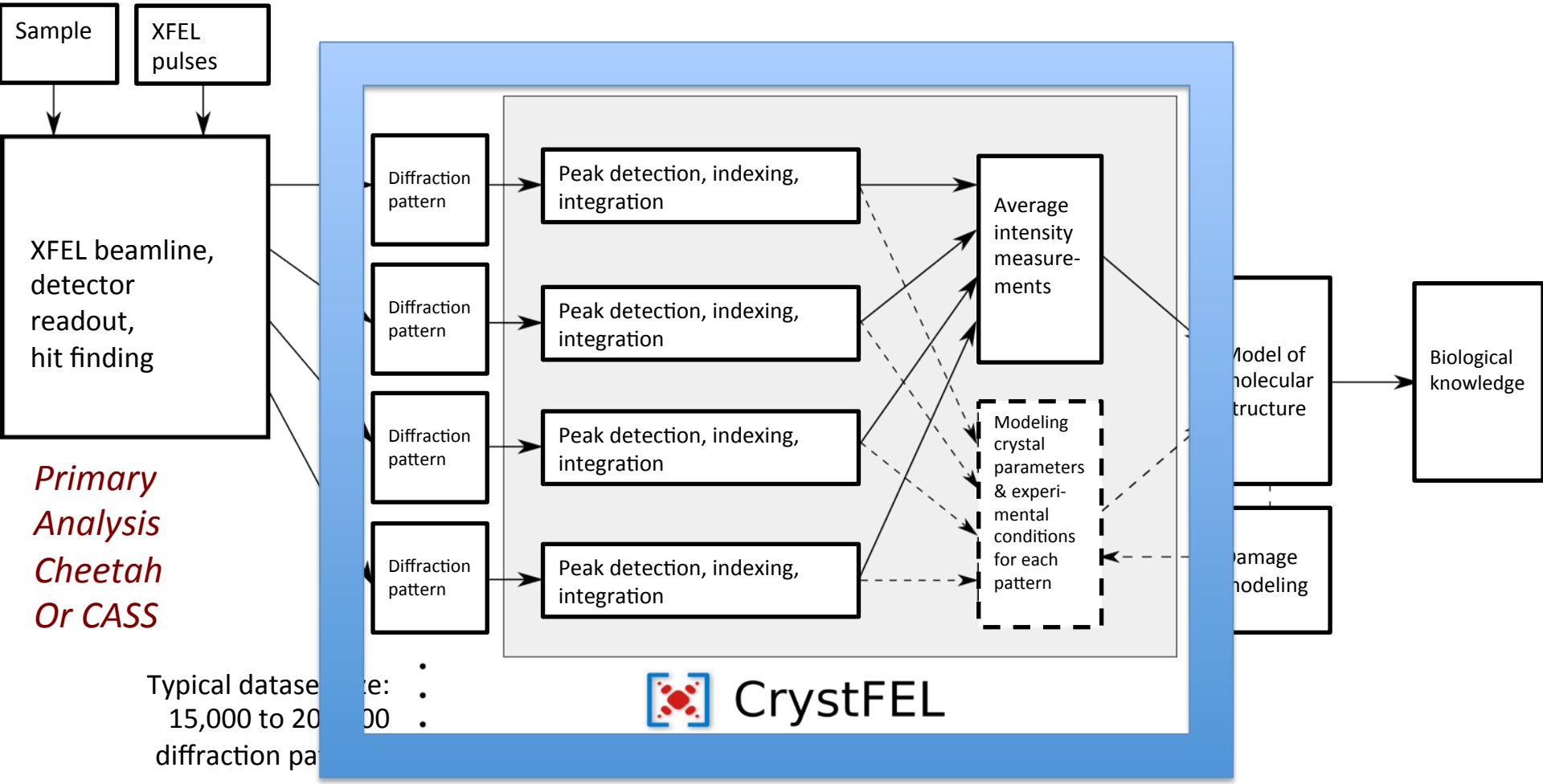
An angular integration over the Bragg spot is needed to obtain the structure factor.

One snapshot gives partial reflections.

SFX data analysis overview

1. Detector calibration – geometry refinement; detector artifact removal
2. Autoindexing - unit cell and orientation determination
3. Integration of reflections - prediction of spots and background subtraction
4. Merging - choosing crystal symmetry; are the data twinned?
5. Scaling – modeling of crystal and experimental parameters; detwinning?
6. Phasing – molecular replacement, refinement, etc.
7. Electron density map

SFX data analysis overview



Primary Analysis Cheetah Or CASS

Typical datase e: 15,000 to 20 00 diffraction pa

Secondary analysis

Figure adapted from Thomas White

Primary data analysis - Cheetah

Raw XFEL data

Cheetah github.com/antonbarty/cheetah

CORRECTIONS
1. Identify saturated pixels
2. Dark current subtraction
3. Correct common mode offset (module wise)
4. Apply gain calibration (pixel wise)
5. Mask bad pixels (hot/cold/dead)

MAIN TASKS
1. **Estimate & subtract of background**
2. Identification and integration of Bragg peaks (see below). Need > 15 spots for indexing.
3. **Hit finding** and frame sorting

PEAK SEARCH
1. Check for pixel intensity above threshold
2. Check if signal to noise (SNR) is above SNR threshold (I/sigma(I)) ← **uses LOCAL background**
3. Find connected pixels that are above ADC threshold
4. Find centroid of peak (x,y)
5. Output peak details

Huge data rates

LCLS runs at 120 Hz (max)
CSPAD (detector at CXI, LCLS)
= 2.3 x 10^6 pixels, each 16 bit
→ 4.6 MB / image
→ 2 TB / hour
→ potentially **120 TB from a single LCLS experiment** (5 shifts)
(or 240 TB when using 2 CSPADs)

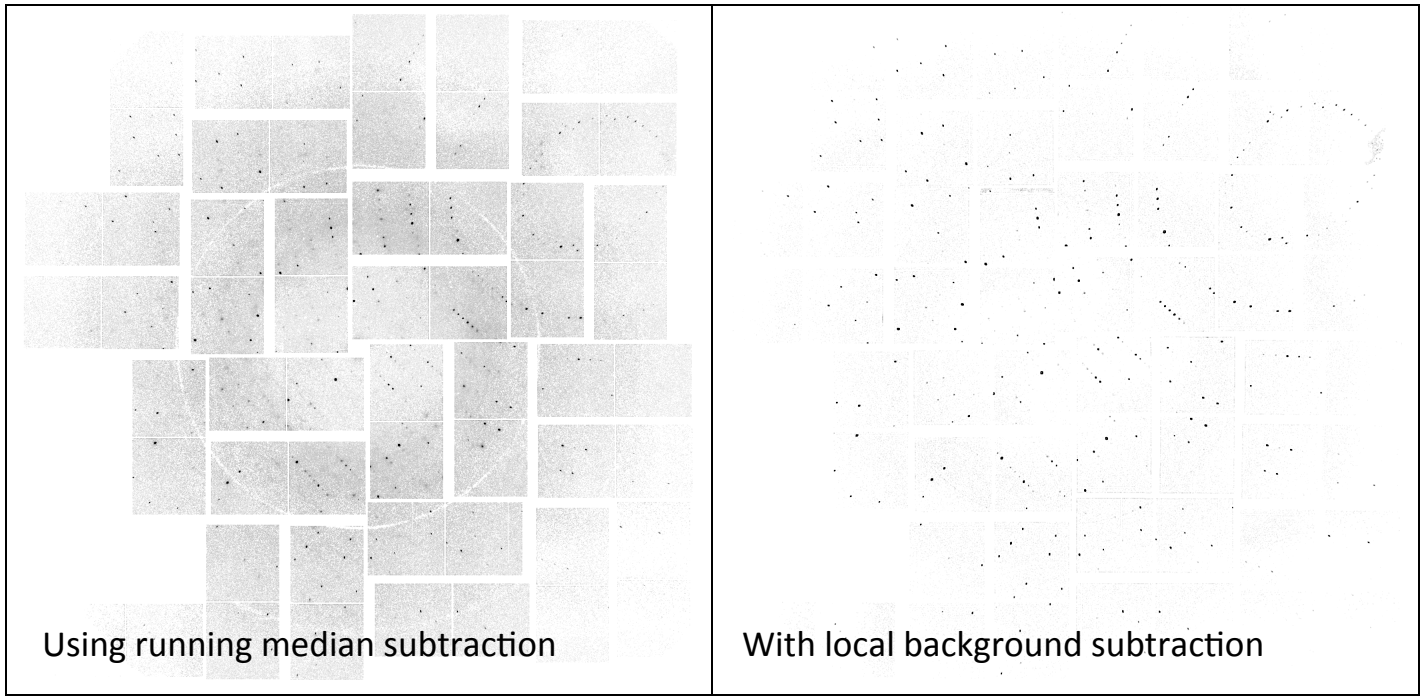
European XFEL will run at 27,000 Hz…

OUTPUT
- **Corrected diffraction images** (hits only, HDF5 format)
- Lists of peaks found
- Radial averages
- Virtual powders (hits, misses)
- Other relevant metadata: λ, detector distance, pump probe signals for time resolved work for example and
- Hit finding statistics
- XFEL pulse spectrum, autocorrelation, etc

Cheetah – background subtraction

- Background subtraction
  - for samples flowing in a liquid or gas stream the femtosecond-duration X-ray pulses capture snapshot images of the background medium on timescales shorter than those of their intrinsic fluctuations. Background subtraction algorithms must take account of these fluctuations.
  - When the photon background is relatively steady from shot to shot but changes slowly over the course of many frames we subtract a background estimated from the recent history of non-hit frames. This “running background” subtraction typically works well for samples in the gas phase, and is relatively efficient to calculate; however it can prove problematic for samples flowing in a liquid suspension where there is significant shot-to-shot variation in background.
  - Liquid jet: local bg subtraction works much better. Use SNR to find peaks. It is optional to save this or to let CrystFEL recalculate the background

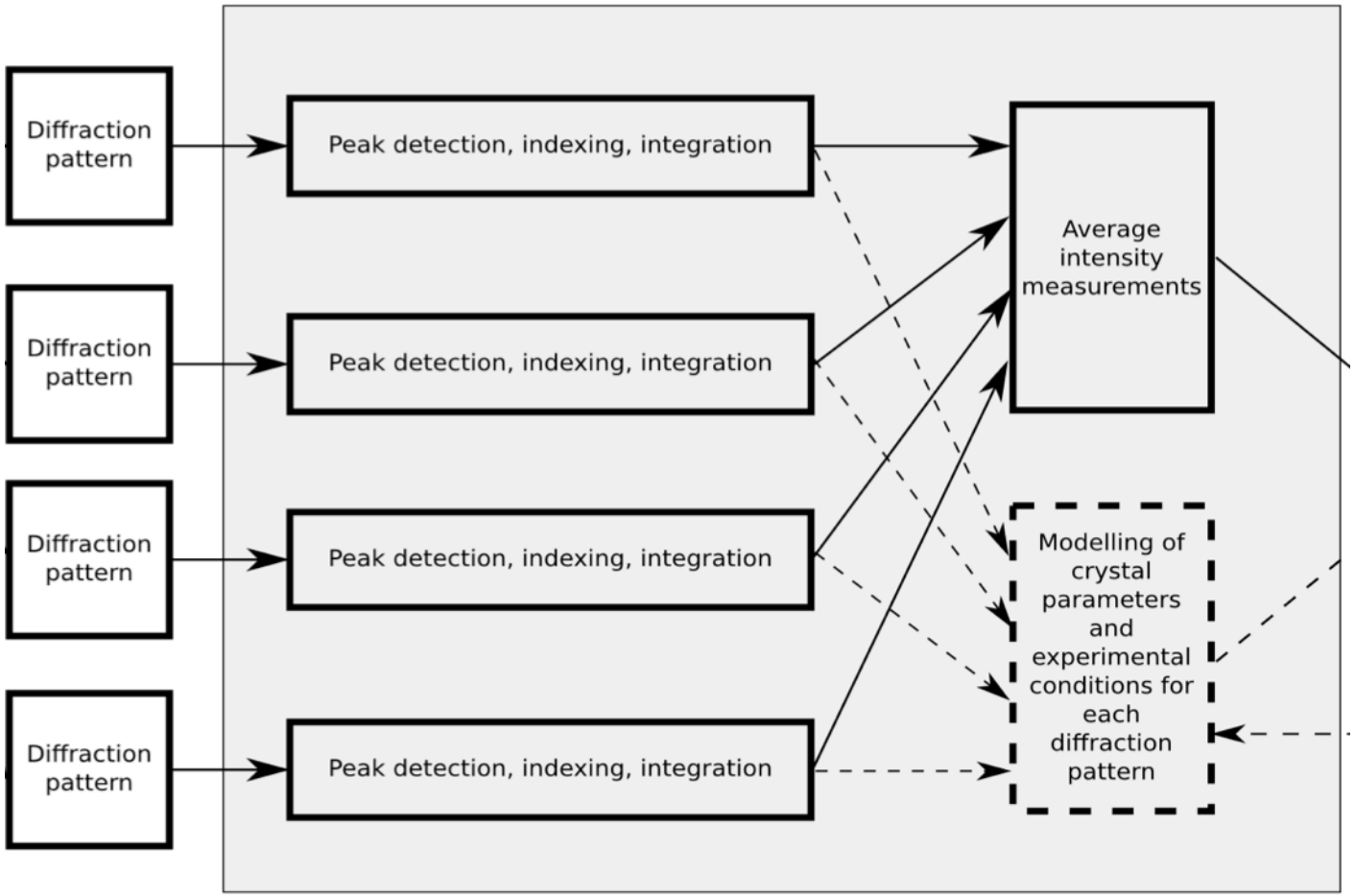
Cheetah – background subtraction



Comparison of results from running background subtraction and local background subtraction for crystalline samples flowing in a water jet. (a) Image after subtraction of water ring averaged over multiple frames; fluctuations in pulse intensity and water jet struture result in imperfect background subtraction using running background subtraction . (b) Subtraction of local background using moving median filter of width 7 pixels produces a cleaner image for peak detection.

Figure from "High throughput analysis of serial X-ray diffraction data from free electron lasers" Anton Barty, Richard Kirian, Chun Hong Yoon, Filipe Maia, Thomas A. White and Henry Chapman.

Our focus: CrystFEL



Peak list (Cheetah or CrystFEL) → autoindexing (e.g. Mosflm doesn't know about the modular detector; CrystFEL handles all of that) → accept solution if constraints are met (e.g. a == b, or angles) → predict spot positions (depends on your beam file) → accept (and integrate) spot if more constraints are met (e.g. integer hkl, overlaps, saturation, background integration possible ..)

Dealing with SFX data volumes

- Very large volumes of data (TBs) & large numbers of diffraction patterns (millions to start with)
- “Hit finding” significantly reduces the amount of data by selecting potentially indexable snapshots by looking for Bragg spots and throwing away empty or very weak shots. Most X-rays don’t hit crystals.
- Images are indexed individually
- Parallel processing is critical for data reduction / and hit finding and indexing (sequential processing would be too slow!)

Secondary data analysis: CrystFEL

- Open source suite of programs created specifically for processing serial crystallography data collected at XFELs – parallel processing
- Constituent programs deal with **viewing, indexing, integrating, merging, scaling** and **evaluating** the quality of the data, and also simulating patterns
- Final output can be fed into CCP4, Phenix etc

- www.desy.de/~twhite/crystfel/
- Main author & architect: Thomas White (CFEL, DESY)
- Contributors: Kenneth Beyerlein, Richard Kirian, Andrew Aquila, Andrew Martin, Lorenzo Galli, Chun Hong Yoon, Nadia Zatsepin, Karol Nass, Anton Barty … and some of you, one day?

CrystFEL: suite of programs for serial crystallography data analysis

Indexamajig: quick indexing and integrating of large numbers of diffraction patterns.

pattern_sim: diffraction pattern simulation.

process_hkl: merging Bragg intensities using the Monte Carlo method.

Extra programs to help with the individual stages of the data analysis:

check_hkl: calculating figures of merit for merged data.

compare_hkl: examining the differences between two sets of merged intensities.

render_hkl: plotting intensities, structure factors and multiplicities in 2D and 3D.

sum_stack: summing diffraction patterns after peak detection to produce a 2D 'virtual powder pattern', which can be used to quickly evaluate the amount of data collected.

powder_plot: summing data from a wider range of formats (image, reflection list or peak-list form) into 1D 'powder' traces.

get_hkl: perform various manipulations on reflection data, such as artificially 'twinning' their intensities, expanding them out to point groups of lower symmetry, adding noise or filtering reflections according to a template file.

hdfsee: view for diffraction patterns (in hdf5 format).

Ongoing development in SFX

- Hit rates are low: waste sample & time
  - Smaller jets? Slow jets, e.g. LCP - lipidic cubic phase?
  - Liquid jet requires milliliters of sample at a few mg/mL
  - LCP requires microliters at a few mg/mL
- Time resolved studies
  - Shot-to-shot fluctuations are bad
  - Using nanocrystals could be really beneficial
- Indexing ambiguities
  - Data from two different crystals may be accidentally merged in twin related orientations.
- Scaling…
  - Need to model crystal parameters and experimental conditions for each pattern

What is CrystFEL?

- Open source software suite created specifically for processing serial crystallography data collected at XFELs
- Constituent programs deal with viewing, indexing, integrating, merging, scaling and evaluating the quality of the data, and also simulating patterns
- www.desy.de/~twhite/crystfel/
- Main author & architect: Thomas White (CFEL, DESY)
- Contributors: Kenneth Beyerlein, Richard Kirian, Andrew Aquila, Andrew Martin, Lorenzo Galli, Chun Hong Yoon, Nadia Zatsepin, Karol Nass, Anton Barty ... you?

Where can I get help with CrystFEL?

- www.desy.de/~twhite/crystfel/

$ indexamajig –h (or indexamajig –help) gives usage instructions & flags

$ man indexamajig

manual pages are available for crystfel, crystlfe_geometry, check_hkl, compare_hkl, get_hkl, hdfsee, indexamajig, partial_sim, martialator, pattern_sim, process_hkl, render_hkl

- White, T. A. et al. (2012). *CrystFEL: a software suite for snapshot serial crystallography*. J. Appl. Cryst. 45, 335–341.
- T. A. White et al. (2013) *Crystallographic data processing for free-electron laser sources.* Acta Cryst D. Accepted.
- Kirian, R. A. et al. (2010) *Femtosecond protein nanocrystallography-data analysis methods*. Optics Express 2010, *18* (6), 5713-5723.

How is each diffraction pattern processed?

Flow diagram of diffraction pattern processing in indexamajig.

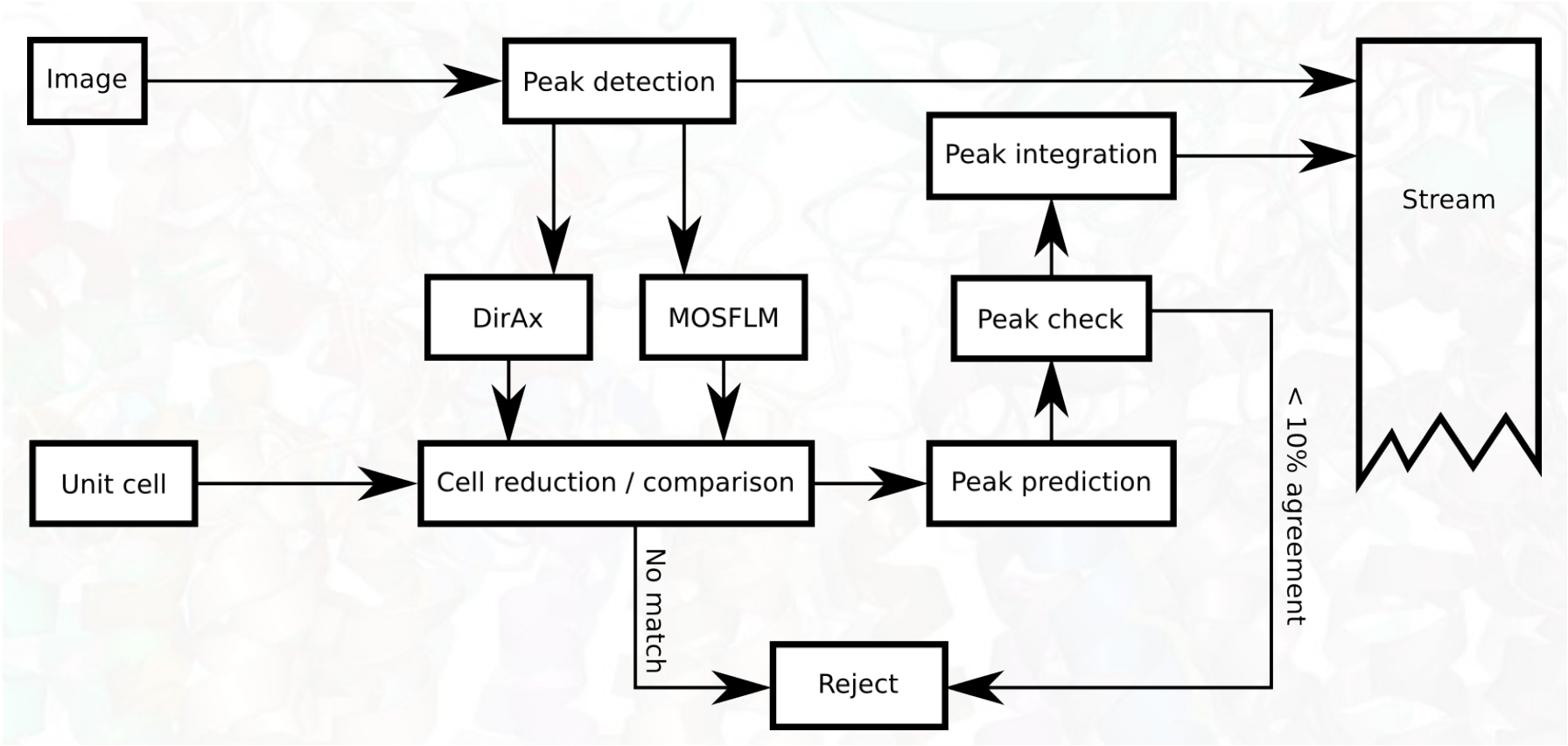


Figure: Thomas White
White, T. A. et al. J. Appl. Cryst. (2012). 45, 335–341

Indexing, merging and evaluation workflow in CrystFEL

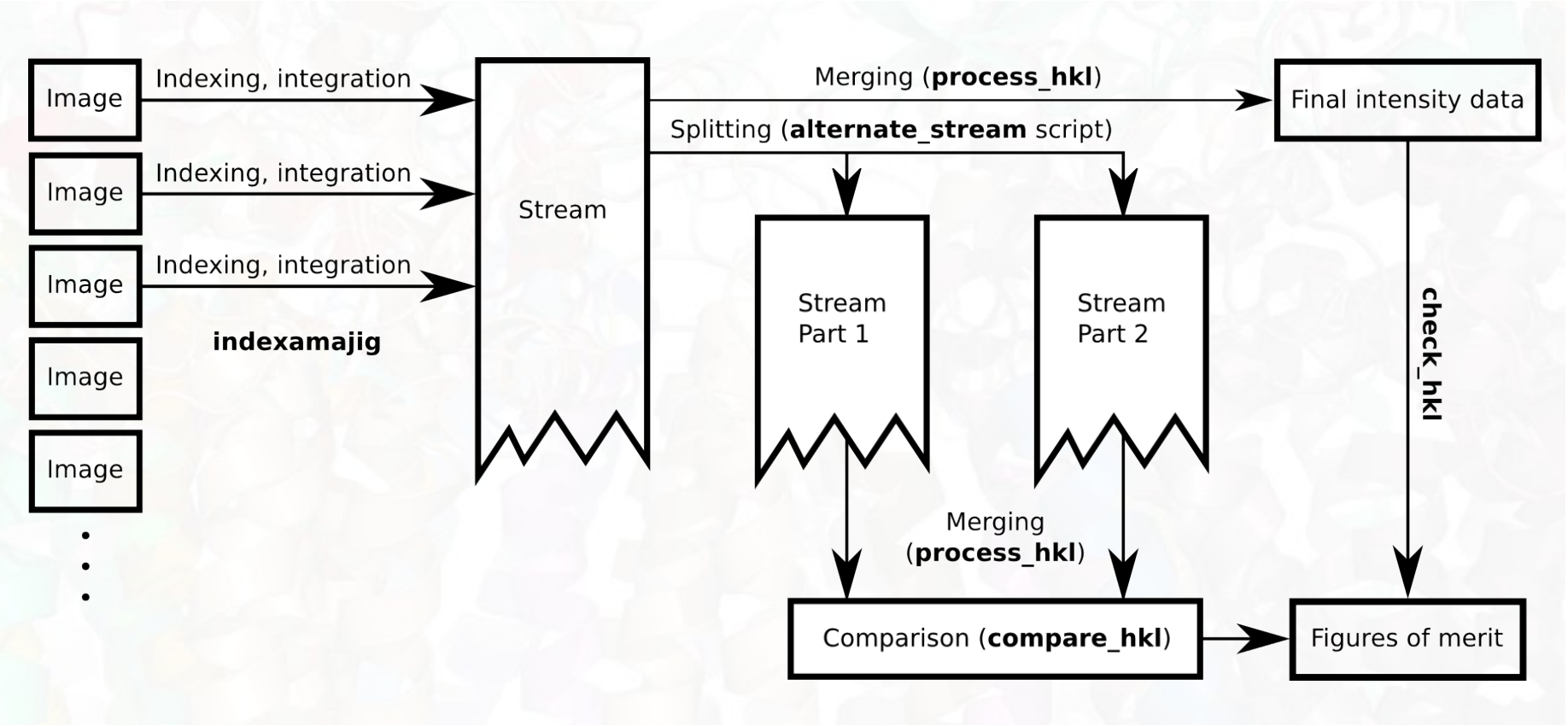


Figure: Thomas White
White, T. A. et al. J. Appl. Cryst. (2012). 45, 335–341

What you need to start indexing

The minimum requirements for indexing with CrystFEL's "indexamajig"

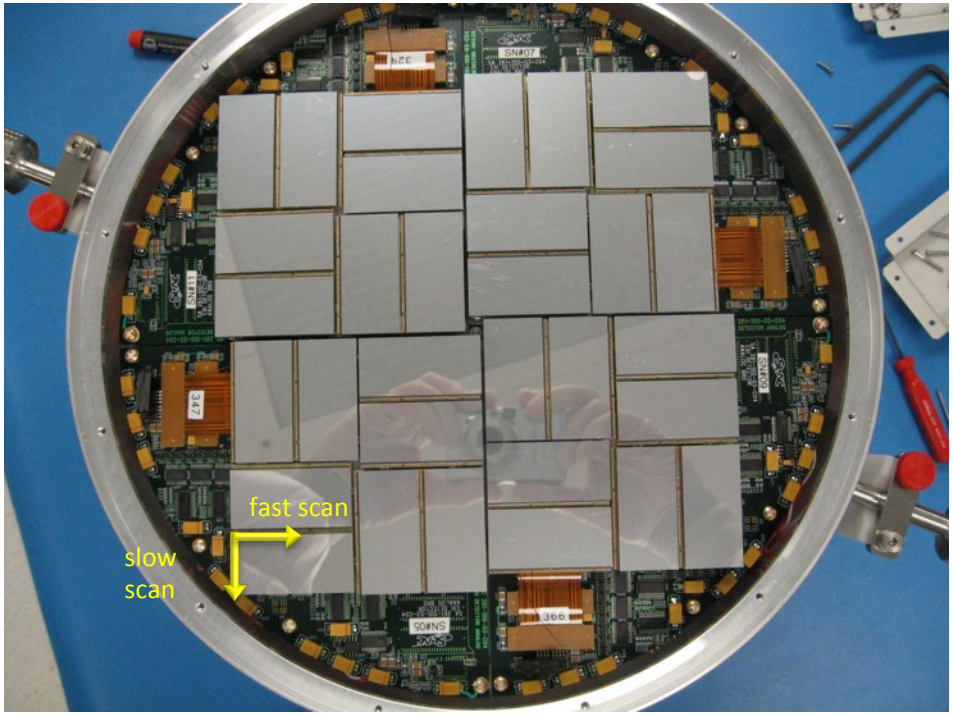
- Diffraction patterns (HDF5 format)
- Geometry file (text file)
- Beam parameter file (text file)

- Optional: unit cell parameters (PDB file "CRYST1" line)

Cornell-SLAC pixel array detector (CSPAD) at CXI, LCLS

- 32 modules (2 ASICs each) tiled to fill about 1700 x 1700 pixels with gaps between modules
- 110 x 110 μm^2 pixels
- 185 x 194 pixels per ASIC (application specific integrated circuit)
- Quadrants are independently movable to change size of hole in center (instead of using a beam stop)
- 2 gain settings (each pixel)
- Signal-to-noise ~ 3.5 (high gain mode)
- Dynamic range of about 350 photons at 9.4 keV
- 120 Hz readout:

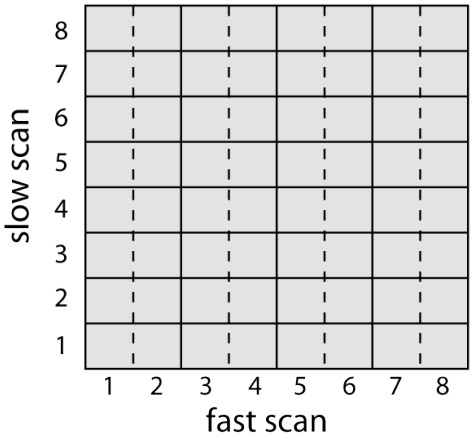
  16 bits * 185 rows * 194 cols * 120Hz * 64 ASICs = 4.4 Gbps per CSPAD



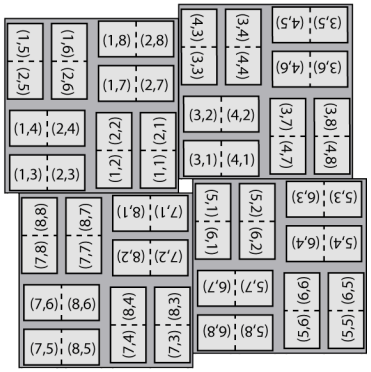
Metrology (and image source): confluence.slac.stanford.edu/display/PCDS/CSPad+metrology+and+calibration+files,+links

H. T. Philipp, M. Hromalik, M. Tate, L. Koerner, S. M. Gruner. (2011) *Pixel array detector for x-ray free electron laser experiments.* Nucl Instrum. Methods A **649**, 67.

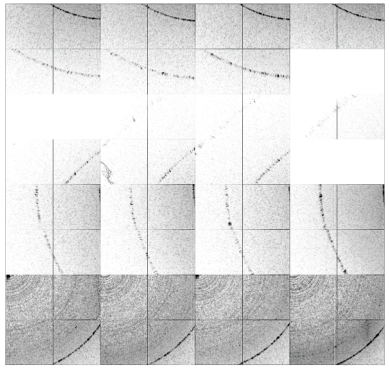
CSPAD geometry



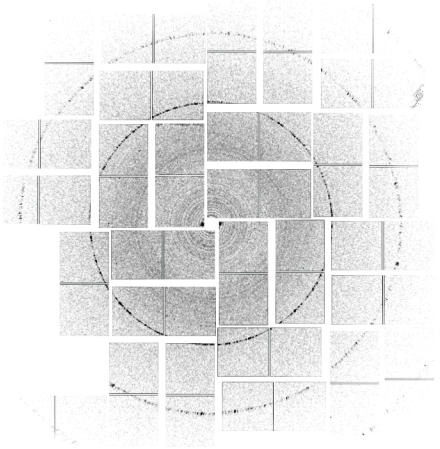
"Raw", non-interpolated layout of detector data



"Assembled" layout of modules corresponding to the physical detector



Virtual powder pattern in the "raw" layout



Virtual powder pattern in the "assembled" layout

CrystFEL geometry file

CrystFEL's geometry file maps the "fast scan" and "slow scan" directions onto physical coordinates (x, y, z). This convention can be used for any number of modules.

```
adu_per_eV = 0.00105
max_adu = 3500
coffset = 2.0e-3

q0a0/min_fs =    0
q0a0/min_ss =    0
q0a0/max_fs =  193
q0a0/max_ss =  184
q0a0/badrow_direction = -
q0a0/res =      9090.91
q0a0/clen = /LCLS/detectorPosition
q0a0/fs = -0.0057225772x +0.9999836087y
q0a0/ss = -0.9999836087x -0.0057225772y
q0a0/corner_x =      424.384
q0a0/corner_y =     -10.6473
q0a0/no_index =       0

q0a1/min_fs =  194
q0a1/min_ss =    0
q0a1/max_fs =  387
q0a1/max_ss =  184
q0a1/badrow_direction = -
q0a1/res =      9090.91
q0a1/clen = /LCLS/detectorPosition
q0a1/fs = -0.0057225772x +0.9999836087y
q0a1/ss = -0.9999836087x -0.0057225772y
q0a1/corner_x =      423.257
q0a1/corner_y =      186.349
q0a1/no_index =       0
```

CrystFEL beam file

; Number of photons per pulse
beam/**fluence** = 1.0e12

; Radius of X-ray beam (m)
beam/**radius** = 10.0e-6

; Photon energy in eV (can specify "9340" eV, but ideally use values in HDF5 file)
beam/**photon_energy** = /LCLS/photon_energy_eV

; Bandwidth: FWHM(wavelength) over wavelength. Note: current simulation code
; just uses a rectangular distribution with this as its (full) width.
beam/**bandwidth** = 0.001

; Beam divergence in radians
beam/**divergence** = 0.008

; Reciprocal space profile radius in m^-1
profile_radius = 0.001e9

CrystFEL analysis steps by program name

- **indexamajig** – index patterns, integrate spots in each image
- **check-peak-detection** (script to check peak finding)
- check-near-bragg (check accuracy of indexing, i.e. prediction of spot positions)
- **process_hkl** - merge same and related reflections
- **check_hkl** - check reflection list
- **compare_hkl** - compare two reflection lists
- Hdfsee – view HDF5 patterns

Test data: SFX data from lysozyme

- www.cxidb.org/id-17.html
- Download “CrystFEL format”. Both cleaned data formats are HDF5, but the ASIC intensities are stored in separate datasets in the .cxi format (e.g. cxidb-17-run0300.tar), so use the CrystFEL format for these exercises (e.g. run0340.tar.gz)
- gunzip run0340.tar.gz
- tar –xvf run0340.tar.gz

Test data: SFX data from lysozyme

- Hen egg white lysozyme. PDBID: 4ET8
- Unit cell: a = b = 79 Å, c = 38 Å, α = β = γ = 90°
- Space group: $P4_32_12$
- Crystal sizes: approx. 1 x 1 x 3 μm^3
- Sample-detector distance: 93 mm
- 9.4 keV, 40 fs X-ray pulses, collected at LCLS in Feb 2011.
- 600 µJ ($4x10^{11}$ photons/pulse), in 10 μm^2 beam focus
- Radiation dose: 33 MGy / crystal
- Published: Boutet et al. (2012) Science 20 362
- From cxidb.org

Indexing options ("indexamajig" flags)

```
indexamajig --indexing=<indexing method> - <conditions>
```

- Refer to man indexamajig
- Algorithms: DirAx, MOSFLM, ReAx, grainspotter, XDS
- Options: raw, axes, comb, bad, nolatt, latt, cell, nocell
- E.g. `--indexing=mosflm-comb-latt`

- We'll just look at MOSFLM and DirAx

Indexamajig: indexing and integrating spots (in 2D)

```
$ indexamajig --input=lyso.list --output=test.stream --
geometry=cspad-geom.geom --beam=lcls-cxi-9keV.beam --pdb=1VDS.pdb
--peaks=hdf5
--indexing=mosflm,dirax --stream=lyso.stream

--tolerance=<a%>,<b%>,<c%>, angles(deg)
--int-radius = inner, middle, outer
```

Test data are in /workshop/zatsepin/hdf5, so to make a list of the HDF5's (e.g. called "lyso.list"), add this to your script that will run indexamajig:

find /workshop/zatsepin/hdf5/ -name *.h5 > lyso.list

Check peak finding

- Copy *check_peak_prediction* to where your stream file is (i.e. the output from *indexamajig*) from the *scripts* directory within your CrystFEL directory. Make it executable (chmod +x check_peak_prediction)
- Run: ./check_peaks_prediction stream_file
- *hdfsee* will then open the images in turn, with found peaks circled
- *hdfsee* flags can be given to *check_peak_prediction* (e.g. –i 20 –b 1)

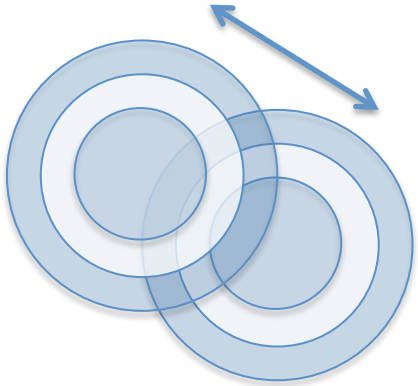
Check Bragg peak prediction

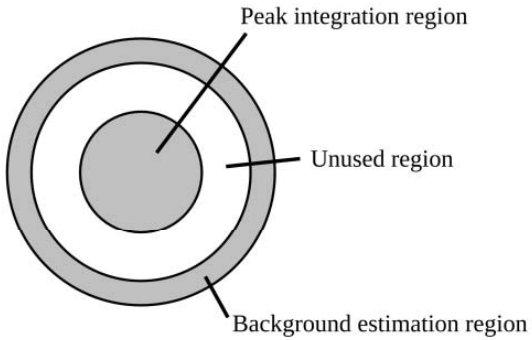
- Copy *check_near_bragg* to where your stream file is (i.e. the output from *indexamajig*) from the *scripts* directory within your CrystFEL directory. Make it executable.
- Run: ./check_near_bragg stream_file
- *hdfsee* will then open the images in turn, with predicted peaks circled
- Consider changing the *profile_radius* in your beam file. Also consider fine-tuning beam/ divergence and beam/bandwidth to more accurately predict only the observed peaks

Integration options in *indexamajig*

- man indexamajig describes the new (mid 2013) integration options, such as centering and 2D profile fitting
- Descriptions and diagrams will be added here later.

Treatment of background , beam divergence and bandwidth in CrystFEL



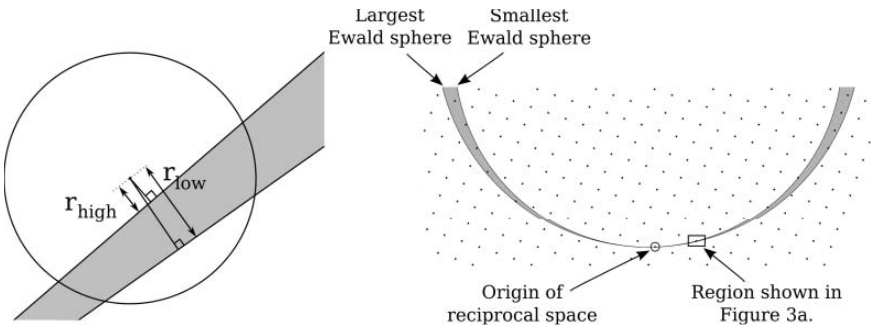


Tom White et al. Acta

$$\sigma^2_{\text{total}} = \sigma^2_{\text{Poisson}} + N\sigma^2_{\text{background}},$$

Fig. 2. Schematic diagram of the peak integration scheme used by CrystFEL.

Large unit cells:
densely packed spots
~6 pixels = too few pixels for an accurate estimate of the background



Recent updates to CrystFEL
Include 2D profile fitting
a la Rossman.
Appl. Cryst. (1979). 12, 225-238

Fig. 3. (a) Geometrical model used for calculation of spot partialties. (b) Context of diagram.

Shape transform is identical around each peak for one nanoxtal, so model it ? Use partialities ?

process_hkl

- process_hkl takes a data stream, such as that from indexamajig, and merges the many individual intensities together to form a single list of reflection intensities which are useful for crystallography.
- Without scaling, the final reflection intensity is the mean of all intensities measured for identical or symmetry-related reflections.

process_hkl

- input: list of integrated reflections (list of hkl's and intensity in each pattern), i.e. the output from indexamajig
- Output: reflection list (h k l Intensity sig(I))

- The output from process_hkl can be converted into a format readable by other (downstream) crystallographic analysis software

```
process_hkl -i test-dm.stream -o test-dm.hkl -y 4/mmm
```

Reflection list quality checks

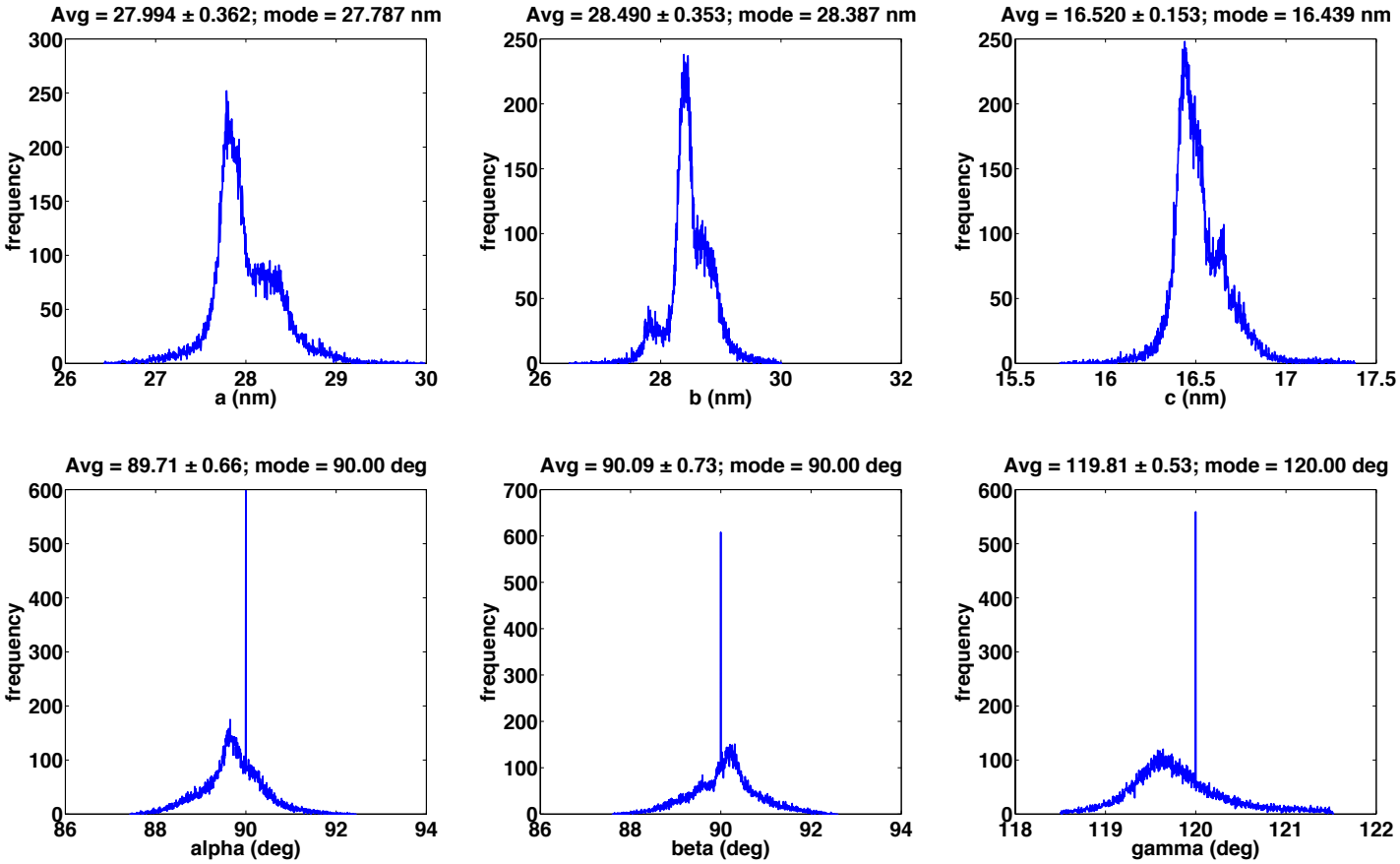
To check the quality of the indexing results, i.e. the stream file (output from **process_hkl**)

- **check_hkl**
  - check_hkl calculates figures of merit for reflection data, such as completeness and average signal strengths, in resolution shells.
  - Needs unit cell (PDB format)
- **compare_hkl**
  - compare_hkl compares two sets of reflection data and calculates figures of merit such as R-factors.
- **cell_please (in crystfel/scripts)**
  - Plots unit cell parameters. Perl script for plotting unit cell histograms with gnuplot

- Convert hkl file to mtz for e.g. phenix.xtriage

Unknown unit cell?

If you don't provide a unit cell and hence accept any indexing solution (e.g. using: --indexing=mosflm-raw), usually peaks corresponding to the right values for a, b, c and angles will stand out, given enough nice patterns



compare_hkl

- compare_hkl compares two sets of reflection data and calculates figures of merit such as R-factors.
- Reflections will be considered equivalent according to your choice of point group.
- Figure of merit options:

$$Rsplit = \frac{1}{\sqrt{2}} \frac{\sum_{hkl} \left| I_{hkl}^{even} - I_{hkl}^{even} \right|}{\frac{1}{2} \sum_{hkl} \left| I_{hkl}^{even} + I_{hkl}^{even} \right|}$$

$$R1f = \frac{\sum_{hkl} \sqrt{I_{hkl}^{even}} - \sqrt{kI_{hkl}^{odd}}}{\sum_{hkl} \sqrt{I_{hkl}^{even}}}$$

$$R1i = \frac{\sum_{hkl} I_{hkl}^{even} - kI_{hkl}^{odd}}{\sum_{hkl} I_{hkl}^{even}}$$

$$R2 = \sqrt{\frac{\sum_{hkl} \left(I_{hkl}^{even} - kI_{hkl}^{odd} \right)^2}{\sum_{hkl} \left(I_{hkl}^{even} \right)^2}}$$

CC - the Pearson correlation coefficient

CC* - See Karplus and Diederichs, Science 336 (2012) p1030.

CCano - The correlation coefficient of the Bijvoet differences of acentric reflections.

I1 and I2 are the intensities of the same reflection in both reflection lists. The scale factor, k, is given by sum(I^{even}*I^{odd}) / sum($I^{odd\ 2}$), unless you use -u.

1. Run "alternate_stream" (in crystfel/scripts) on test.stream file to split it into two stream files: odd and even indexed frames.
2. Run process_hkl on each stream
3. Run compare_hkl on

```
compare_hkl  odd.hkl even.hkl \
        --fom=Rsplit \
        -p lysozyme.pdb \
        --shell-file=shells-Rsplit.dat" \
        -y <pointgroup> \
        --rmax=4.0e9 \
        --rmin=0.8e9 \
        --nshells=15
```

R factors: checking reflection list

$$Rsplit = \frac{1}{\sqrt{2}} \frac{\sum_{hkl} \left| I_{hkl}^{even} - I_{hkl}^{even} \right|}{\frac{1}{2} \sum_{hkl} \left| I_{hkl}^{even} + I_{hkl}^{even} \right|}$$

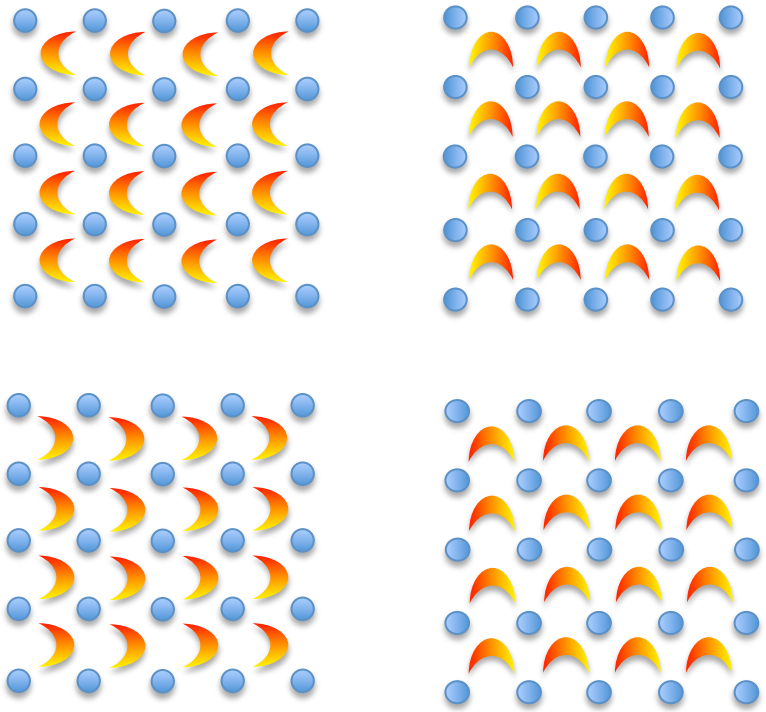
$$R1f = \frac{\sum_{hkl} \sqrt{I_{hkl}^{even}} - \sqrt{kI_{hkl}^{odd}}}{\sum_{hkl} \sqrt{I_{hkl}^{even}}}$$

$$R1i = \frac{\sum_{hkl} I_{hkl}^{even} - kI_{hkl}^{odd}}{\sum_{hkl} I_{hkl}^{even}}$$

$$R2 = \sqrt{\frac{\sum_{hkl} \left(I_{hkl}^{even} - kI_{hkl}^{odd} \right)^2}{\sum_{hkl} \left(I_{hkl}^{even} \right)^2}}$$

Indexing ambiguities

- Crystal lattice is invariant with respect to twinning operator
- The crystal is NOT invariant with respect to twinning operator
- In other words**: the lattice has the same geometry in various orientations of the crystal**
- Autoindexing algorithms can't tell these apart, so SFX data are merged in higher symmetry



Merohedral twinning

- Resolving the indexing ambiguity is hard, but not impossible.

CrystFEL development

- For anyone who wants the cutting edge features with concomitant bugs and documentation defects, get the git repository:

  $ **git clone git://git.bitwiz.org.uk/crystfel.git**

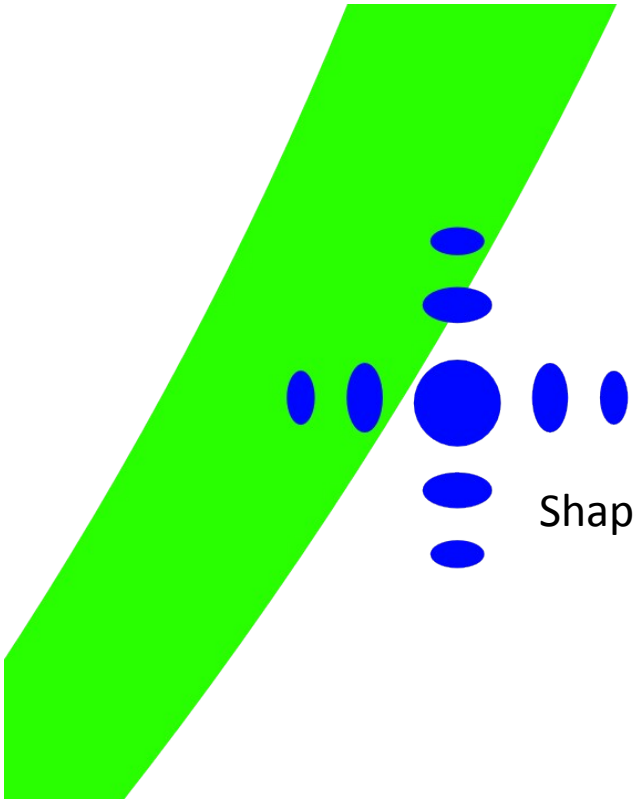
  - written in C with supporting perl and bash scripts
  - released under v. 3 of GNU General Public license
- We rely on bug reports from the community
- Fully supported and documented (see handout, CrystFEL papers, <program> -h, man pages and CrystFEL website)
- www.desy.de/~twhite/crystfel/

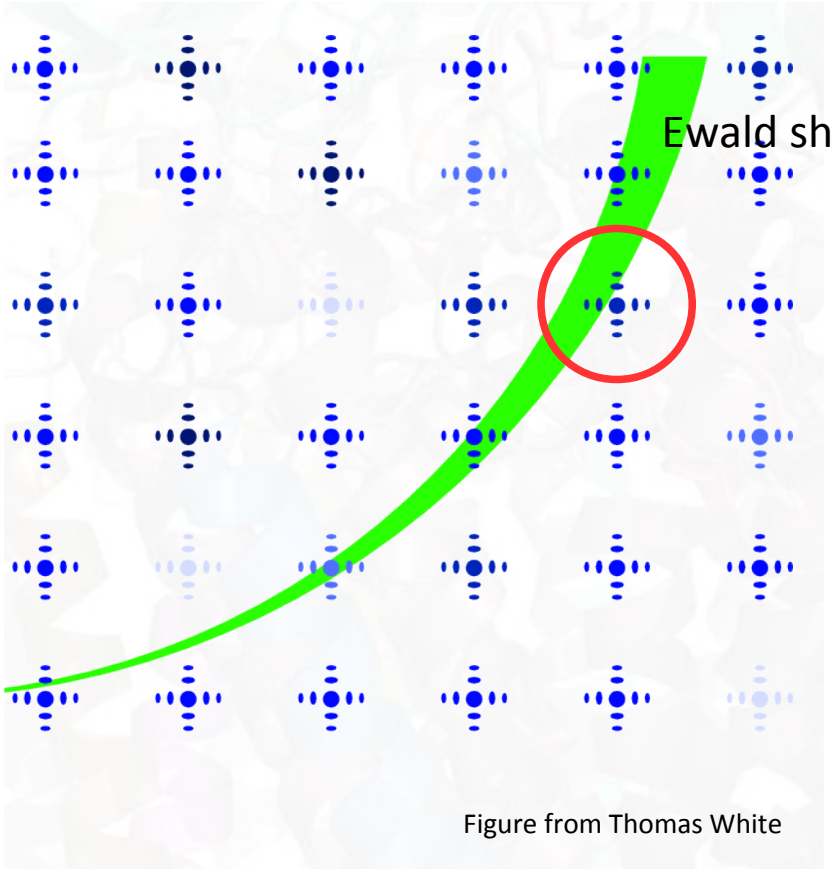
- **Sign up for the CrystFEL mailing list**: email **sympa@desy.de** with the subject: **SUBSCRIBE cfel-cxi-crystfel**
- Alternatively, just email Thomas White: thomas.white@desy.de
- T. A. White et al. "*CrystFEL:* a software suite for snapshot serial crystallography". J. Appl. Cryst. **45**, p335–341.
- http://journals.iucr.org/d/issues/2013/07/00/ba5187/ba5187.pdf

Simple approach to scaling patterns

1. Merge all indexed patterns
    1. Simply add intensities of corresponding reflections
    2. Use symmetry to average symmetry related reflections
2. Go back to pattern 1.
    1. Compare reflections in pattern 1 with the average intensities for each reflection. E.g. {0 0 1} = 10, {0 0 2} = 20, while the average {0 0 1} = 100 and average {0 0 2} = 150. Meaning the scale to bring pattern 1 close to "average" is between 7.5 and 10.
    2. Using sum of RMSD for all reflections in pattern 1, calculate scaling factor (single number for pattern 1) that would bring the reflection intensities in pattern 1 closest to their individual average intensities (from the first merge)
3. Sum all "scaled" patterns to yield new "average" intensities. Multiple rounds of scaling can be performed – hopefully converging to the right relative intensities.
4. These steps do not account for partiality.

Measuring partial reflections

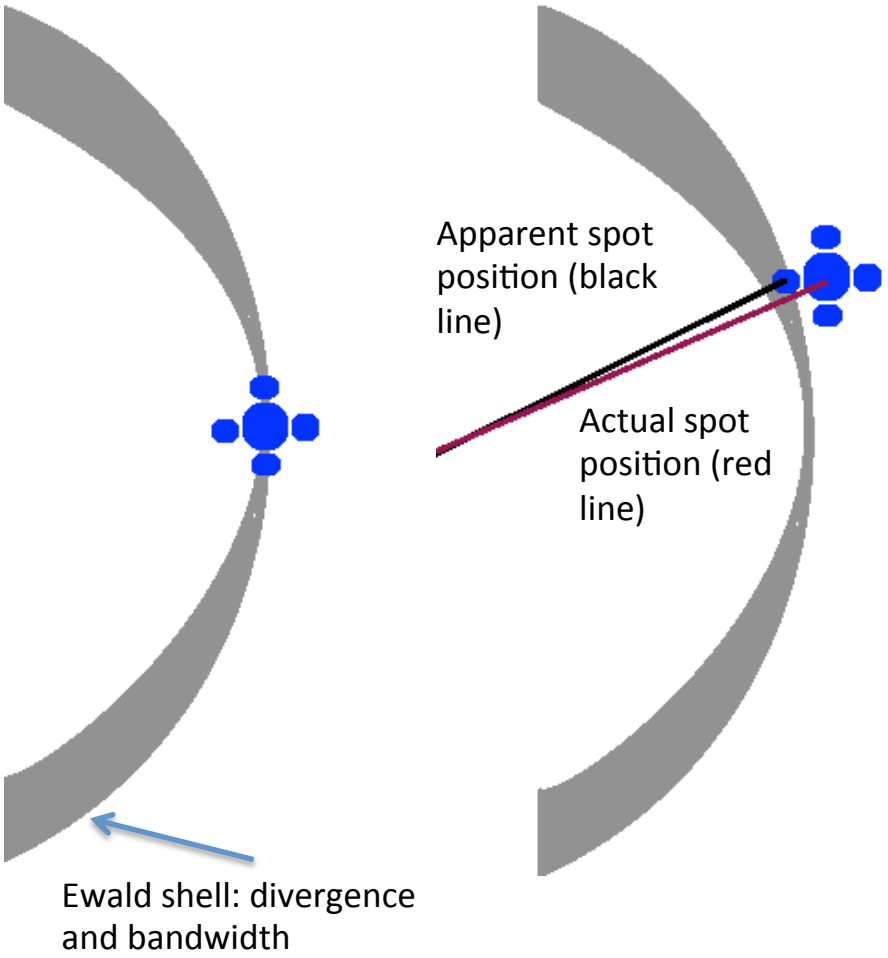




Small crystals + narrow Ewald shell → almost every Bragg spot is only partially recorded. Size effects dominate mosaicity.

Spot prediction

- Shape transforms from nanocrystals complicate prediction of low order reflections
- We don't know every single nanocrystal size, nor their mosaicity, so the size and shape of the Bragg spots are impossible to predict accurately on an individual level
- Extrapolating from (or scaling) partial reflection is possible (but difficult), but dense spots make this even harder because of a higher chance of slight misindexing



Beyond Monte Carlo: scaling of partials

- Fox and Holmes (1966) method:
  Set up full non-linear least squares for the scale
- factors and solve to get the best ones.
- Computational time: **number of patterns squared.**
- Kabsch (2010) method:
  Combine everything with equal scale factors, then
- scale everything to the result, and repeat.
- Computational time: **number of patterns**

Symmetry Classification for Serial Crystallography Experiments

Groups with white backgrounds are merohedral and will exhibit indexing ambiguities. Chiral groups are shown in bold, centrosymmetric groups are underlined.

Move downwards or follow grey arrows to find supergroups which can be accessed with only rotation operations. Do not cross vertical or thick black horizontal lines unless following a grey arrow. When you reach a cell with a shaded background, you have found the corresponding "source symmetry". A partial ambiguity resolution could be attempted into any intermediate group you can reach.

Point Groups		**Space Groups**	
Triclinic lattice			
1	$\underline{\bar{1}}$	**P1**	$\underline{P\bar{1}}$
Monoclinic lattice			
	m		Pm, Pc, Cm, Cc
2	$\underline{2/m}$	**P2, P2$_1$, C2**	$\underline{P2/m}$, $\underline{P2_1/m}$, $\underline{C2/m}$, $\underline{P2/c}$, $\underline{P2_1/c}$, $\underline{C2/c}$
Orthorhombic lattice			
	mm2		Pmm2, Pmc2$_1$, Pcc2, Pma2, Pca2$_1$, Pnc2, Pmn2$_1$, Pba2, Pna2$_1$, Pnn2, Cmm2, Cmc2$_1$, Ccc2, Amm2, Aem2, Ama2, Aea2, Fmm2, Fdd2, Imm2, Iba2, Ima2
222	$\underline{mmm}$	**P222, P222$_1$, P2$_1$2$_1$2, P2$_1$2$_1$2$_1$, C222$_1$, C222, F222, I222, I2$_1$2$_1$2$_1$**	$\underline{Pmmm}$, $\underline{Pnnn}$, $\underline{Pccm}$, $\underline{Pban}$, $\underline{Pmma}$, $\underline{Pnna}$, $\underline{Pmna}$, $\underline{Pcca}$, $\underline{Pbam}$, $\underline{Pccn}$, $\underline{Pbcm}$, $\underline{Pnnm}$, $\underline{Pmmn}$, $\underline{Pbcn}$, $\underline{Pbca}$, $\underline{Pnma}$, $\underline{Cmcm}$, $\underline{Cmce}$, $\underline{Cmmm}$, $\underline{Cccm}$, $\underline{Cmme}$, $\underline{Ccce}$, $\underline{Fmmm}$, $\underline{Fddd}$, $\underline{Immm}$, $\underline{Ibam}$, $\underline{Ibca}$, $\underline{Imma}$
Tetragonal lattice			
4	$\bar{4}$	**P4, P4$_1$, P4$_2$, P4$_3$, I4, I4$_1$**	P$\bar{4}$, I$\bar{4}$
	$\bar{4}$2m, $\bar{4}$m2, $\underline{4/m}$		P$\bar{4}$2m, P$\bar{4}$2c, P$\bar{4}$2$_1$m, P$\bar{4}$2$_1$c, I$\bar{4}$2m, I$\bar{4}$2d; P$\bar{4}$m2, P$\bar{4}$c2, P$\bar{4}$b2, P$\bar{4}$n2, I$\bar{4}$m2, I$\bar{4}$c2; $\underline{P4/m}$, $\underline{P4_2/m}$, $\underline{P4/n}$, $\underline{P4_2/n}$, $\underline{I4/m}$, $\underline{I4_1/a}$
	4mm		P4mm, P4bm, P4$_2$cm, P4$_2$nm, P4cc, P4nc, P4$_2$mc, P4$_2$bc, I4mm, I4cm, I4$_1$md, I4$_1$cd
422	$\underline{4/mmm}$	**P422, P42$_1$2, P4$_1$22, P4$_1$2$_1$2, P4$_2$22, P4$_2$2$_1$2, P4$_3$22, P4$_3$2$_1$2, I422, I4$_1$22**	$\underline{P4/mmm}$, $\underline{P4/mcc}$, $\underline{P4/nbm}$, $\underline{P4/nnc}$, $\underline{P4/mbm}$, $\underline{P4/mnc}$, $\underline{P4/nmm}$, $\underline{P4/ncc}$, $\underline{P4_2/mmc}$, $\underline{P4_2/mcm}$, $\underline{P4_2/nbc}$, $\underline{P4_2/nnm}$, $\underline{P4_2/mbc}$, $\underline{P4_2/mnm}$, $\underline{P4_2/nmc}$, $\underline{P4_2/ncm}$, $\underline{I4/mmm}$, $\underline{I4/mcm}$, $\underline{I4_1/amd}$, $\underline{I4_1/acd}$
Rhombohedral lattice			
3	$\underline{\bar{3}}$	**R3 (H3)**	$\underline{R\bar{3}}$ ($\underline{H\bar{3}}$); R3m (H3m), R3c (H3c)
32	$\underline{\bar{3}m}$	**R32 (H32)**	$\underline{R\bar{3}m}$ ($\underline{H\bar{3}m}$), $\underline{R\bar{3}c}$ ($\underline{H\bar{3}c}$)

http://www.desy.de/~twhite/crystfel/twin-calculator.pdf

Hexagonal lattice

3			$\bar{3}$								
6	312	321	3m1	$\bar{6}$	31m	$\bar{3}$m1	$\bar{6}$m2	$\bar{6}$2m	$\bar{3}$1m	6/m	6mm
622			6/mmm								

P3, P3_1, P3_2			P$\bar{3}$								
P6, P6_1, P6_5, P6_2, P6_4, P6_3	P312, P$3_1$12, P$3_2$12	P321, P$3_1$21, P$3_2$21	P3m1, P3c1	P$\bar{6}$	P31m, P31c	P$\bar{3}$m1, P$\bar{3}$c1	P$\bar{6}$m2, P$\bar{6}$c2	P$\bar{6}$2m, P$\bar{6}$2c	P$\bar{3}$1m, P$\bar{3}$1c	P6/m, P6_3/m	P6mm, P6cc, P6_3cm, P6_3mc
P622, P$6_1$22, P$6_5$22, P$6_2$22, P$6_4$22, P$6_3$22			P6/mmm, P6/mcc, P6_3/mcm, P6_3/mmc								

Cubic lattice

23	$\bar{4}$3m	m$\bar{3}$	P23, F23, I23, P$2_1$3, I$2_1$3	P$\bar{4}$3m, F$\bar{4}$3m, I$\bar{4}$3m, P$\bar{4}$3n, F$\bar{4}$3c, I$\bar{4}$3d	Pm$\bar{3}$, Pn$\bar{3}$, Fm$\bar{3}$, Fd$\bar{3}$, Im$\bar{3}$, Pa$\bar{3}$, Ia$\bar{3}$
432	m$\bar{3}$m		P432, P$4_2$32, F432, F$4_1$32, I432, P$4_3$32, P$4_1$32, I$4_1$32	Pm$\bar{3}$m, Pn$\bar{3}$n, Pm$\bar{3}$n, Pn$\bar{3}$m, Fm$\bar{3}$m, Fm$\bar{3}$c, Fd$\bar{3}$m, Fd$\bar{3}$c, Im$\bar{3}$m, Ia$\bar{3}$d	

Laue Classes

$\bar{1}$	1			
2/m	2	m		
mmm	222	mm2		
4/m	4	$\bar{4}$		
4/mmm	422	$\bar{4}$2m	$\bar{4}$m2	4mm

$\bar{3}$	3	
$\bar{3}$m	32	3m
$\bar{3}$m1	321	3m1
$\bar{3}$1m	312	31m

6/m	6	$\bar{6}$		
6/mmm	622	$\bar{6}$m2	$\bar{6}$2m	6mm
m$\bar{3}$	23			
m$\bar{3}$m	432	$\bar{4}$3m		

http://www.desy.de/~twhite/crystfel/twin-calculator.pdf

A CrystFEL GUI is on its way!

CrystFEL assistant