

27 July 2012, Harvard Medical School



Partnering Data Collection and Reduction in the Beamline Environment

Diffraction Integration for Advanced Light Sources: West I

Nicholas Sauter

*Computational Crystallography Initiative
Lawrence Berkeley National Laboratory*

Today's presenters:

Graeme Winter, Diamond Light Source

David Waterman, CCP4

Jon Schuermann, NE-CAT

David Neau, NE-CAT

Chris Nielson, ADSC

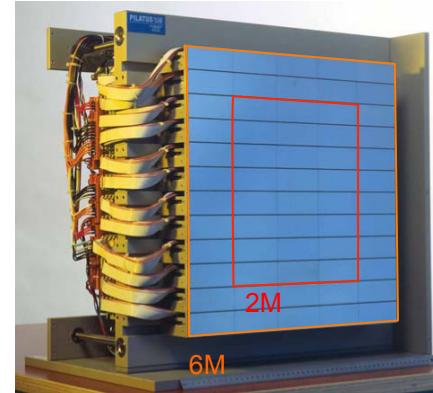
Christian Broennimann, Dectris, Ltd.

Herbert Bernstein, Dowling College

Sandor Brockhauser, ESRF

Pixel array detectors are revolutionizing the experiment

- Synchrotron-based
 - Raster screens
 - Fine slicing



Dectris Ltd.

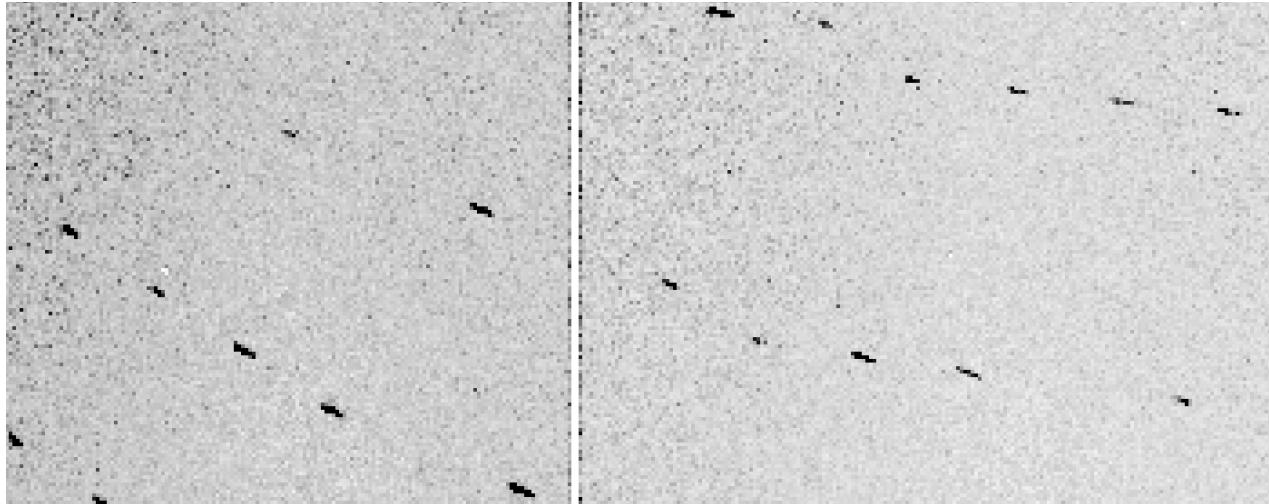
- X-ray free-electron lasers
 - Serial femtosecond crystallography
 - In-situ crystals
 - Single particles



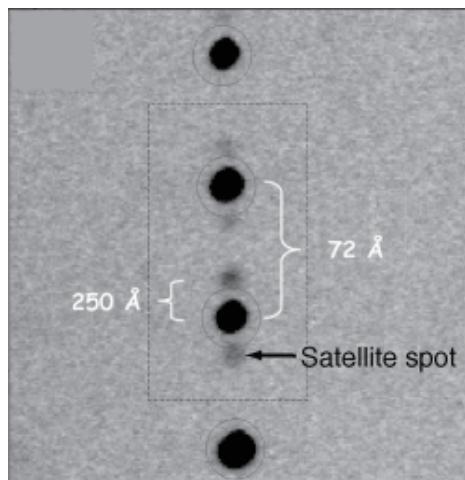
Sol Gruner (2010) IEEE Trans Nuc Sci, 57: 3795

Detailed modeling of diffraction phenomena

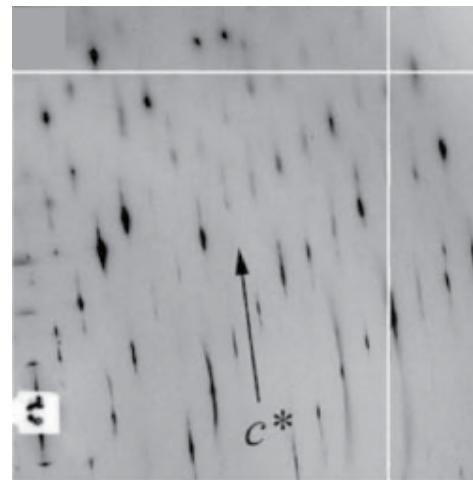
- Bragg spot size and shape



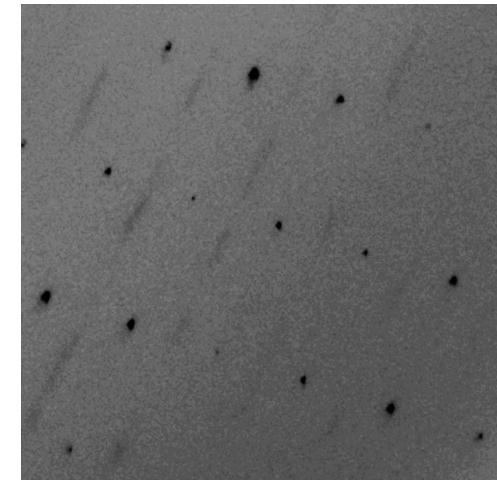
- Lattice phenomena



Incommensurate modulation
Lovelace (2008) J Appl Cryst 41: 600
Porta (2011) Acta D67: 628

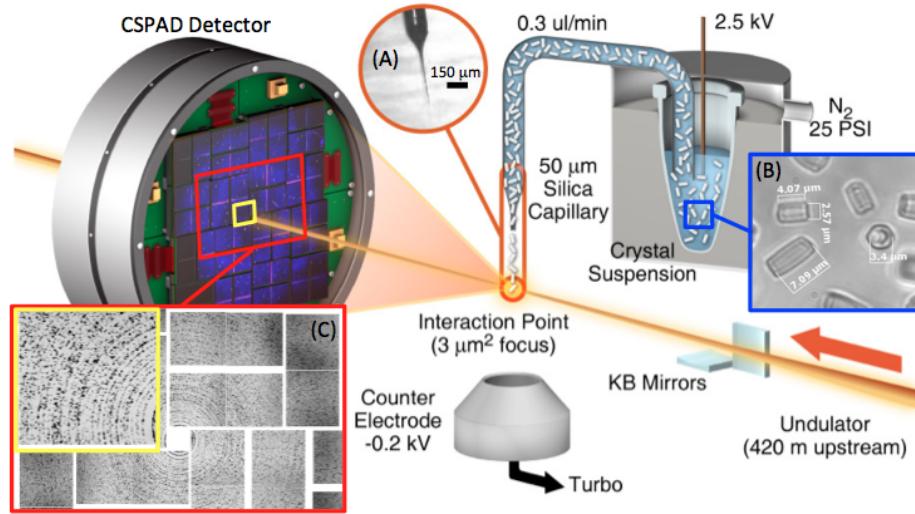


Lattice-translocation disorder
Yeates (2009) Acta D65: 980

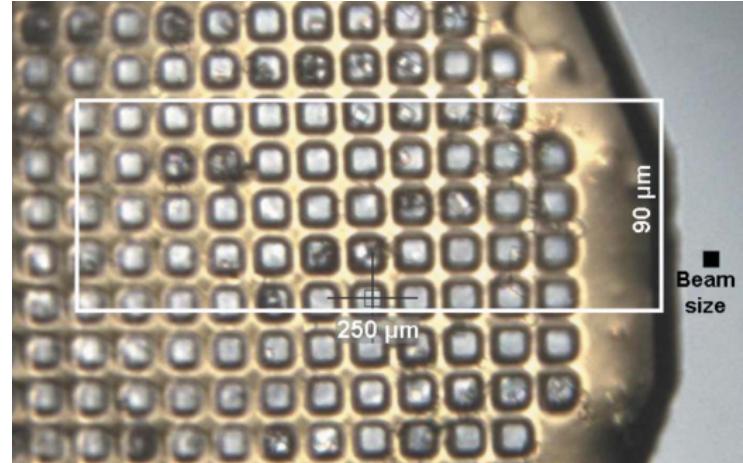


Order-disorder twinning
Dauter (2009) Acta D65: 906

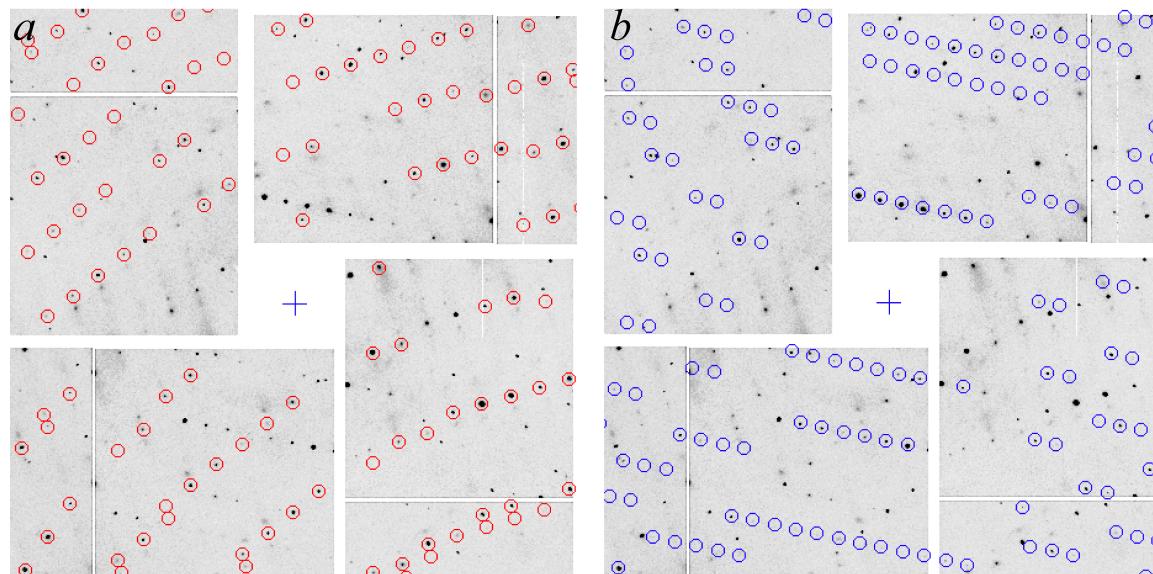
Observation of multiple lattices



Michael Bogan, SLAC



Aishima (2010) Acta D66:1032



DIALS: Diffraction Integration for Advanced Light Sources

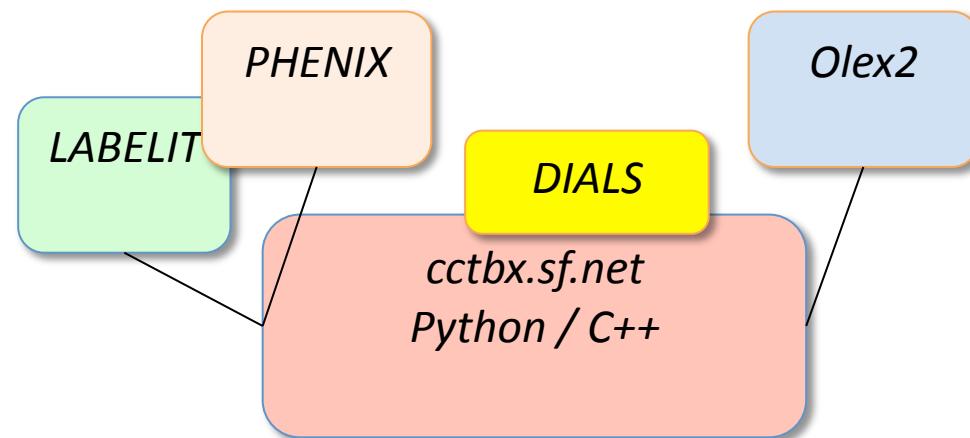
Software collaboration between LBNL and BioStruct-X

Common aims, separate grant proposals and funding sources:

- LBNL, Nick Sauter: NIGMS *R01-GM095887*
- EU, Gwyndaf Evans: BioStruct-X Work Package 6: contract 283570

General agreement on software management

- Open source; BSD-style license to allow distribution of derivative works
- Others can modify source code & contribute to the official version
- Manage and disseminate the improvements from others



CCTBX (Computational Crystallography Toolbox) & data reduction

LABELIT

cctbx.sf.net
Python / C++

Data Formats

ADSC, SMV
Rigaku, Raxis, D*Trek
Bruker
Rayonix, marIP
MacScience
Hamamatsu
CBF, Pilatus miniCBF

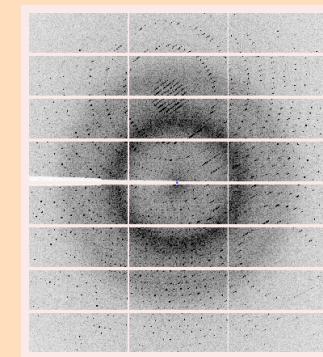
```
Image = ImageFactory(filename)
Spots = Image.get_spotfinder()
Tiles = Image.get_tile_manager()
Graphics = Image.get_flex_image()
```

Arrays
Linear algebra
Structure factors
Unit cells
Space group symmetry

```
from mod_python import apache
log = spotfinder.signal_strength()
request.write(log)
spotfinder
return apache.OK
```

```
#pyana: XFEL data stream
class mod_spotfinder:
    def beginjob():
    def event():
    def endjob()
```

wxPython



pyCUDA

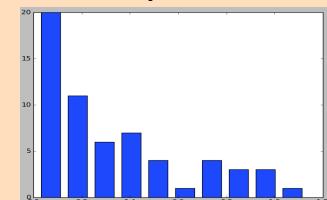
Univ. of Maryland
Approximate
Nearest Neighbor library

CBF lib

HDF5

ccp4:
MTZ lib

matplotlib



Code repository at Sourceforge



SourceForge.net > Find Software > [cctbx](#) > SCM Repositories > [cctbx](#) > trunk

SCM Repositories - [cctbx](#)

Files shown: 1

Directory revision: [15541](#) (of [15541](#))

Sticky Revision:

File	Rev.	Age	Author	Last log entry
Parent Directory				
boost_adaptbx/	15506	4 days	natechols	testing tweaks - make sure temporary file names are unique
cbflib_adaptbx/	14766	4 months	nksauter	Bug fix; found better place to free up FILE* when an exception is thrown.
cctbx/	15539	25 hours	natechols	test changes
chiltbx/	9625	2 years	rwgk	global SConscript changes: env_base now pre-defines CCFLAGS, CXXFLAGS, SHCCFLAGS...
clipper_adaptbx/	12843	14 months	rwgk	preparation for -Qnew
cma_es/	14682	4 months	rjgildea	cma_es/cma_es_ext.cpp: order of includes matters on OSX (at with least with Pyth...
compcomm/	13956	8 months	pjuhas	Removed executable permissions from files without hashbang line. This prevents t...
cootbx/	15274	2 months	natechols	oops
crys3d/	15458	2 weeks	natechols	unicode fix
cudatbx/	15533	46 hours	bkpoon	Templated structure_factor_kernel
dox/	14752	4 months	rwgk	dox/rst/versions.txt: newsletter contents; svn mailing list
dox.sphinx/	15257	2 months	jamasi	add documentation coverage check
fable/	15119	2 months	rwgk	fable/fem/write.hpp: dynamic determination of write_crlf, to fix problem with lb...
fftw3tbx/	13956	8 months	pjuhas	Removed executable permissions from files without hashbang line. This prevents t...
gltbx/	15487	9 days	natechols	test refactoring to support new parallel execution framework
iotbx/	15540	24 hours	natechols	bug fix

Toolkit collaboration: Nightly build & test, platforms, bundles

The screenshot shows a web browser window with the title "cctbx Multi-platform build results". The address bar contains the URL "http://cci.lbl.gov/cctbx_build/". The page content includes a navigation bar with links like "House Winter Allotment", "Gmail", "Web-Ice wiki", "FAQ about C...n telescope", "BCSBWebice", "Yahoo", "Google Maps", "Lawrence Be... Laboratory", and "Popular". Below the navigation bar, there is a section titled "cctbx Multi-platform build results".

cctbx Automatic multi-platform builds

All [cctbx](#) modules are automatically assembled into self-contained source code bundles and self-extracting binary bundles for a variety of platforms. The *Build tags* indicate the date and time (PST or PDT) when the build was initiated.

[\[Change logs\]](#)

Note: builds with tags in parentheses are unreleased and may be removed at any time.
[\[Hide unreleased builds\]](#)

Local build times:

(2012_06_08_0807) [\[summary\]](#)

(2012_06_06_1000) [\[summary\]](#)

(2012_06_04_2102) [\[summary\]](#)

(2012_06_03_1140) [\[summary\]](#)

(2012_06_01_2335) [\[summary\]](#)

(2012_06_01_0047) [\[summary\]](#)

(2012_05_30_2324) [\[summary\]](#)

(2012_05_28_2348) [\[summary\]](#)

(2012_05_26_1126) [\[summary\]](#)

(2012_05_24_0234) [\[summary\]](#)

(2012_05_23_0029) [\[summary\]](#)

(2012_05_20_2215) [\[summary\]](#)

(2012_05_20_0308) [\[summary\]](#)

(2012_05_18_0242) [\[summary\]](#)

(2012_05_16_2203) [\[summary\]](#)

Build tag: 2012_06_06_1000

Self-extracting binary distributions

- cctbx+Python bundles are larger but completely self-contained.
- cctbx only bundles use the Python that ships with the OS.
- cctbx plus includes Python, Numpy, Reportlab, Image Library, wxPython, matplotlib, and HDF5
- Use [stream] if your browser displays the [direct] download in a text window instead of saving it to a file.
- Use [cctbx.python](#) to run scripts. The plain `python` command is no longer supported to avoid confusion with the Python version provided by operating systems.

Platform	cctbx+Python	cctbx only	cctbx plus
CentOS 5.2 Python 2.7.3	[stream] [direct] 45.3 MB		[download] 163.4 MB
CentOS 5.5 (x86_64) Python 2.7.3	[stream] [direct] 49.8 MB		[download] 173.4 MB
Fedora 11 (x86_64) Python 2.7.3	[stream] [direct] 47.6 MB		[download] 168.5 MB
Fedora 12 (x86_64) Python 2.7.3	[stream] [direct] 47.8 MB		[download] 171.2 MB
Fedora 13 (x86_64)	Python 2.6.4 ships with OS	[stream] [direct] 26.6 MB	
Fedora 14 (x86_64)	Python 2.7 ships with OS	[stream] [direct] 26.3 MB	

Build tag: 2012_06_06_1000

Error messages from all log files

Number of build logs: 25
Number of lines with error messages or warnings build+test logs: 0
The number of OK and u+s lines is not the same on all platforms!
(OK, u+s) pair counts: (323, 105): 1, (337, 105): 24

[centos52_py273_build_log](#) [\[folded\]](#)

Total time build: 184.95 seconds [factor 4.12]

[centos52_py273_test_log](#) [\[folded\]](#)

Number of OK & u+: 337, 105

Total time tests: 5287.10 seconds [factor 3.41]

[centos55_x86_64_py273_build_log](#) [\[folded\]](#)

Total time build: 53.42 seconds [factor 1.19]

[centos55_x86_64_py273_test_log](#) [\[folded\]](#)

Number of OK & u+: 337, 105

Total time tests: 2543.14 seconds [factor 1.64]

```
libtbx.python "/net/worm/scratch1/builder/sources/libtbx/tst_stl_utils.py"
OK

libtbx.python "/net/worm/scratch1/builder/sources/libtbx/table_utils.py"
OK

libtbx.python "/net/worm/scratch1/builder/sources/libtbx/tst_dlite.py"
OK

libtbx.python "/net/worm/scratch1/builder/sources/libtbx/phill/tst_tokenizer.py"
OK

libtbx.python "/net/worm/scratch1/builder/sources/libtbx/phill/tst.py"
OK

libtbx.python "/net/worm/scratch1/builder/sources/libtbx/phill/tst_experimental.py"
OK
```

```
libtbx.python "/net/worm/scratch1/builder/sources/libtbx/tst_stl_utils.py"
OK

libtbx.python "/net/worm/scratch1/builder/sources/libtbx/tst_dlite.py"
OK

libtbx.python "/net/worm/scratch1/builder/sources/libtbx/phill/tst_tokenizer.py"
OK

libtbx.python "/net/worm/scratch1/builder/sources/libtbx/phill/tst.py"
OK

libtbx.python "/net/worm/scratch1/builder/sources/libtbx/phill/tst_experimental.py"
OK
```

```
libtbx.python "/net/worm/scratch1/builder/sources/libtbx/tst_stl_utils.py"
OK

libtbx.python "/net/worm/scratch1/builder/sources/libtbx/tst_dlite.py"
OK

libtbx.python "/net/worm/scratch1/builder/sources/libtbx/phill/tst_tokenizer.py"
OK

libtbx.python "/net/worm/scratch1/builder/sources/libtbx/phill/tst.py"
OK

libtbx.python "/net/worm/scratch1/builder/sources/libtbx/phill/tst_experimental.py"
OK
```

Software needs to be fast for high volume work

Scope: Potentially be 1000 images to score @ 0.2sec/image

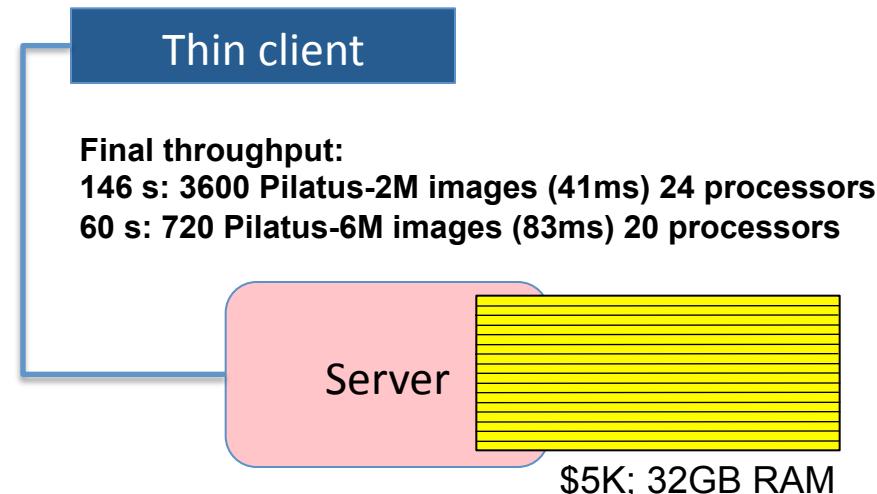
The normal spotfinding process takes ~3 seconds

- 0.5 sec Loading the program
- 0.7 sec Network file read
- 1.5 sec Classify pixels & apply heuristics

Solution: Multiprocessing server handles concurrent requests.

Sauter (2011) *Comp. Cryst. Newslett.* **2**: 93.

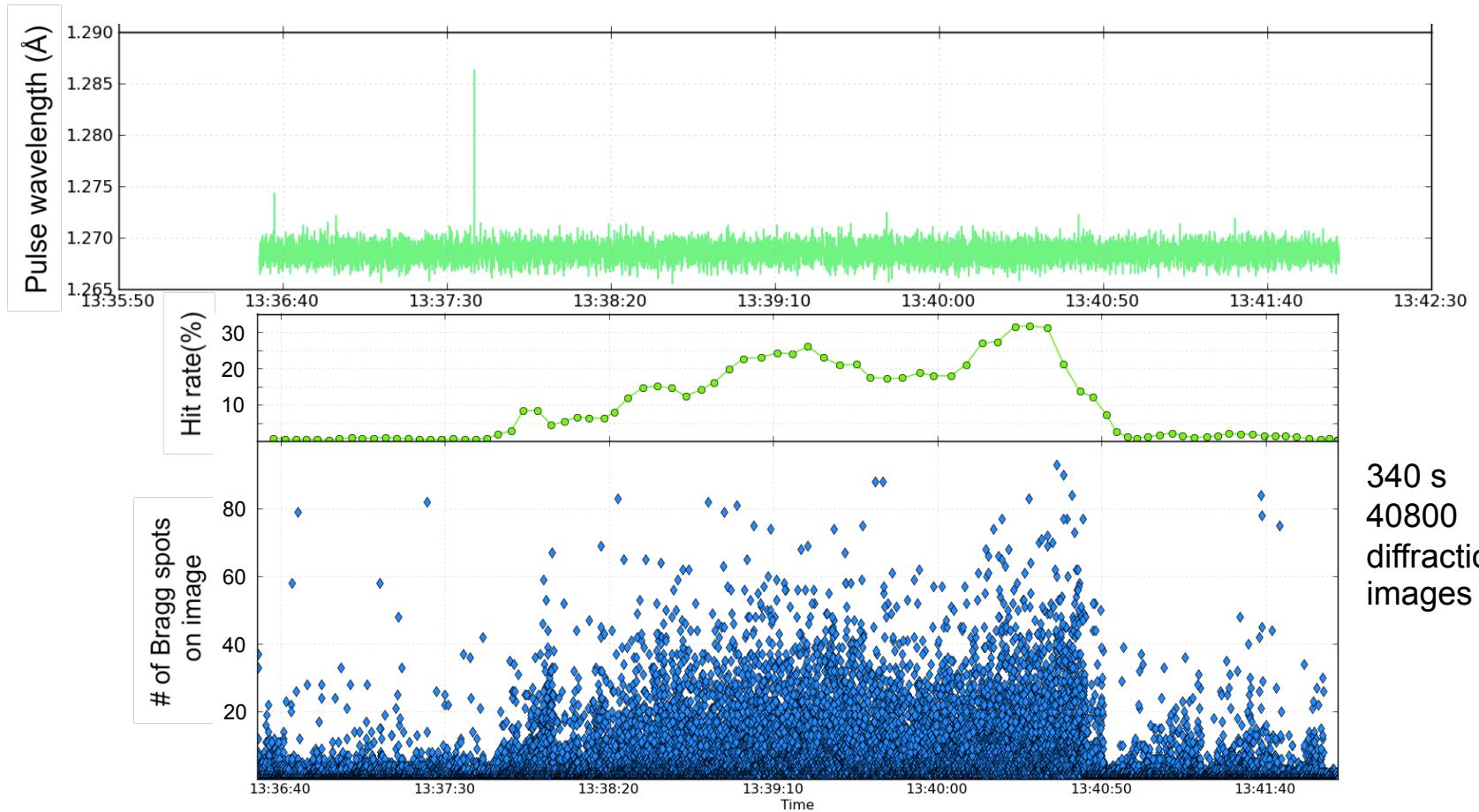
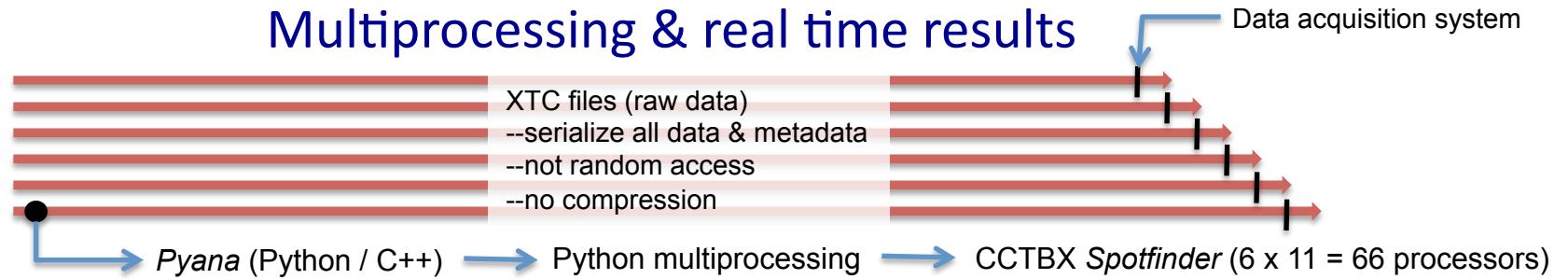
```
from mod_python import apache
log = spotfinder.signal_strength()
request.write(log)
return apache.OK
```



http://localhost:8125/spotfinder/dist1.signal_strength_bcsb?dist1.image=I3_1_0001.cbf

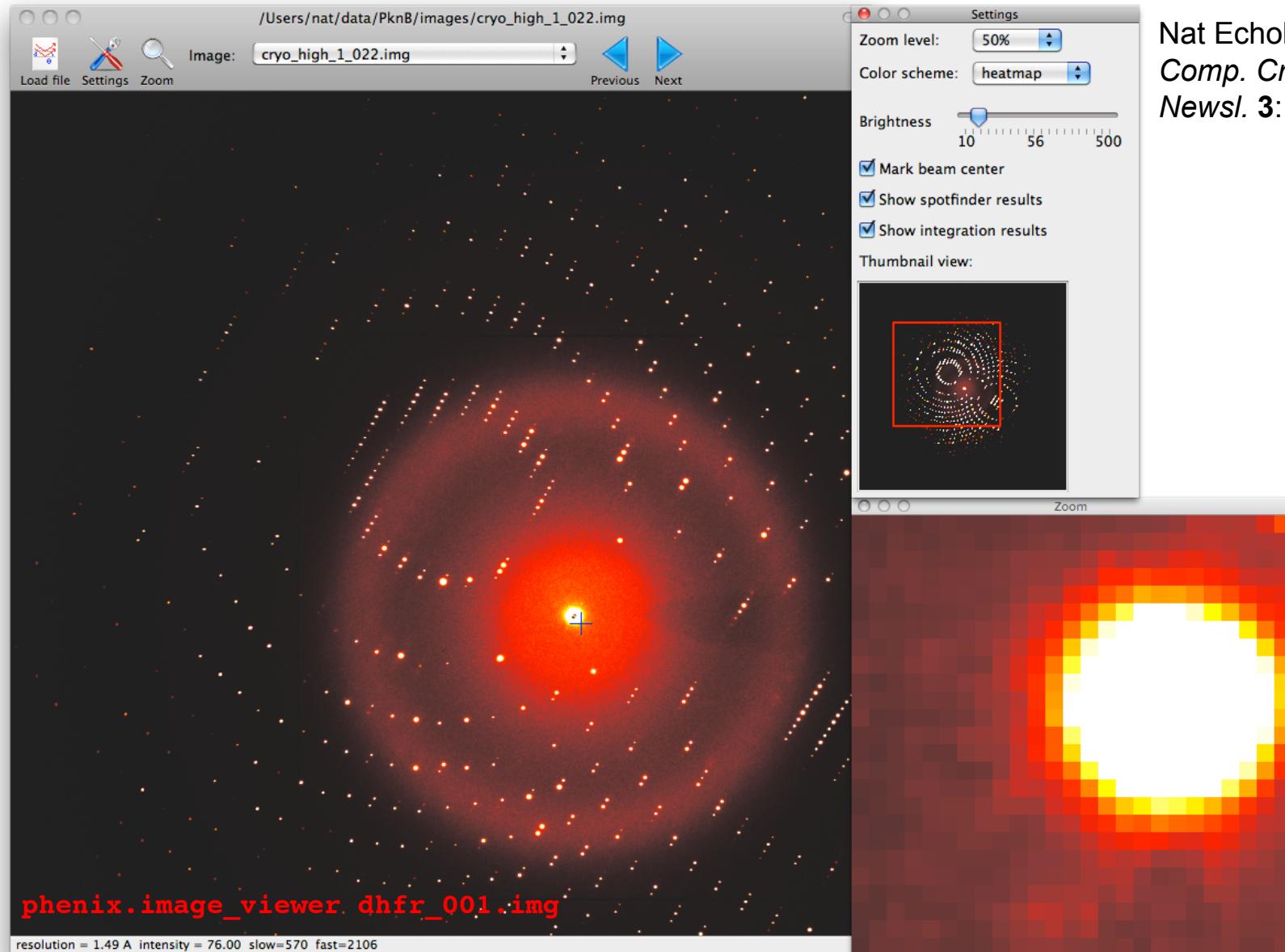
```
<spotfinder>
  <file_name> I3_1_0001.cbf </file_name>
  <total_spots> 710 </total_spots>
  <good_spots> 639 </good_spots>
  <resolution> 1.487 </resolution>
  <mean_isigi> 25.749 </mean_isigi>
  <integrated> 740028.667 </integrated>
  <status> OK </status>
</spotfinder>
```

Multiprocessing & real time results



...Although we interact with the “offline” file-based data, we can still display experimental progress in near-real-time...Hitfinder is complete after 620 seconds.

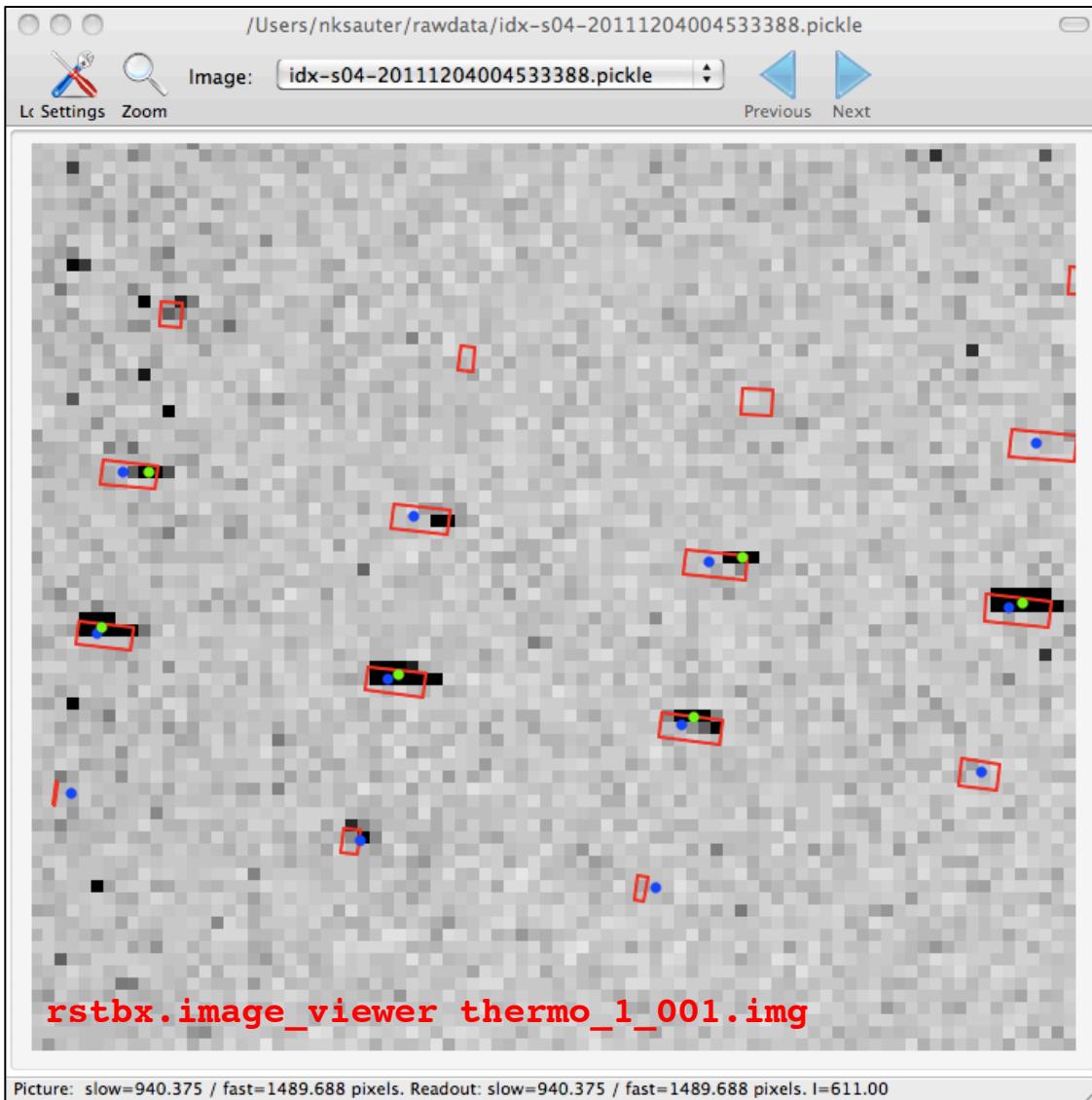
Python-based image viewer



Nat Echols (2012)
Comp. Cryst.
NewsL 3:14.

...can be accessed from the PHENIX interface, *or from the open source CCTBX download*

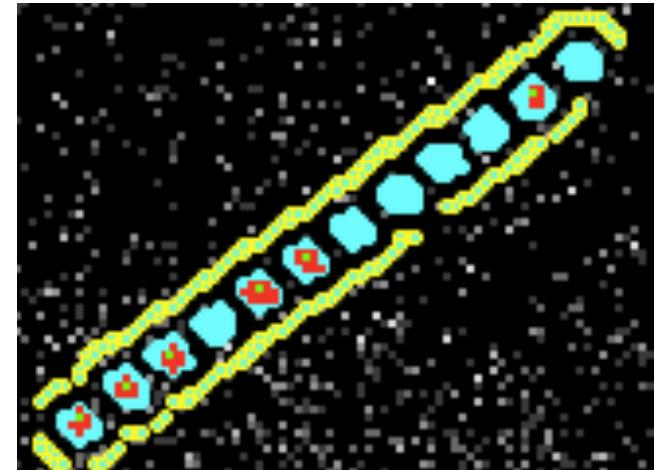
Slip viewer & the fleximage class



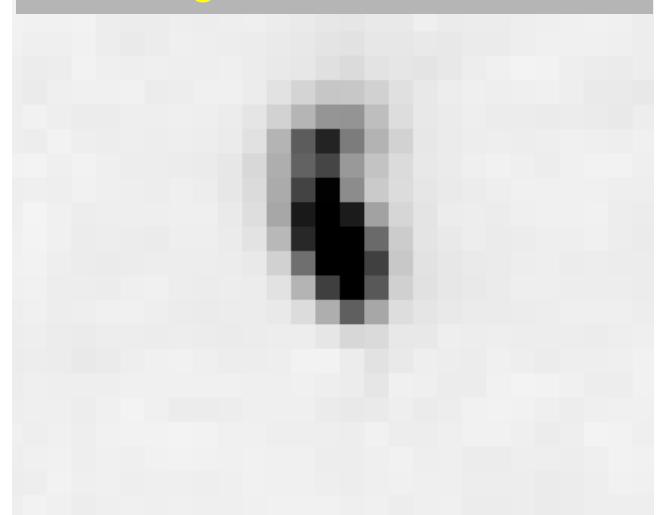
- Developer-oriented colored markup
- Google-like mouse navigation
- Sub-pixel tile precision
- Display tilted detector tiles
- Fast response—cache ahead
- Lab space mapping: represent cylindrical or spherical detectors
- PySlip code—credit to Ross Wilson
<http://code.google.com/p/pyslip>

A Python-based data reduction toolbox

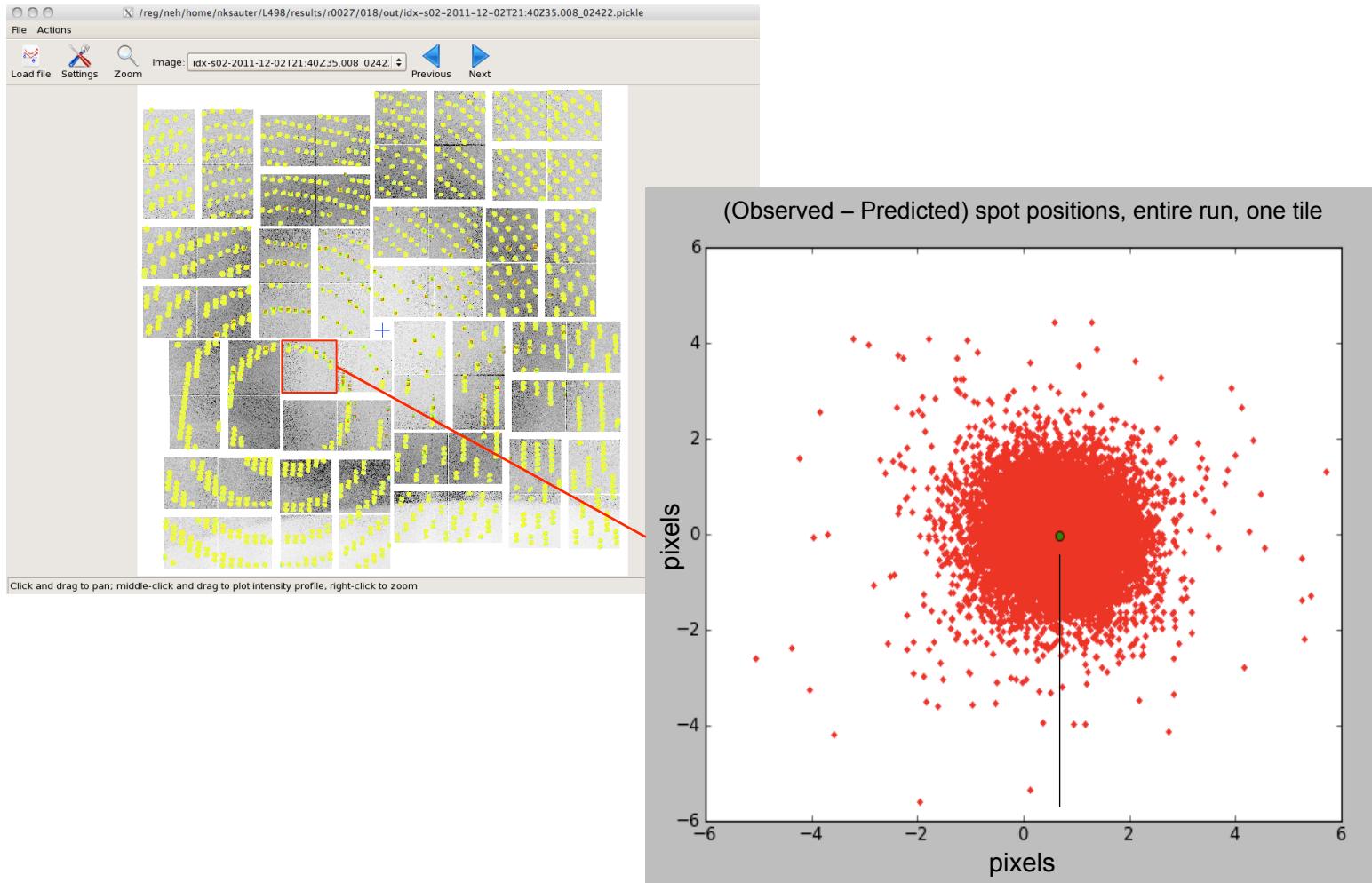
- Ashley Deacon collaboration (2011)
- Pre-existing toolset is extensive
 - spot picking, background fitting (*Spotfinder*)
 - autoindexing (*Labelit*)
 - nearest neighbors (*U. Maryland ANNlib*)
- Initial target:
 - still images, marccd
 - simple integration by standard pixel summation techniques (2D masks)
 - use empirically chosen integration masks from nearby *Spotfinder* spots
 - avoid treating overlaps; integrate only well-separated spots
 - choose same-frame background pixels around the Bragg signal



- *Spotfinder pixels*
- *Signal mask*
- *Background mask*



XFEL data processing: sub-pixel metrology correction



Application to synchrotron datasets

- BioStruct-X WP6 informal collaboration (2012)
 - Graeme Winter (Diamond)
 - David Waterman (CCP4)
- Properly model the rocking curve and Lorentz-polarization corrections:
 - battlefield-tested model (Greenhough & Helliwell, 1982)

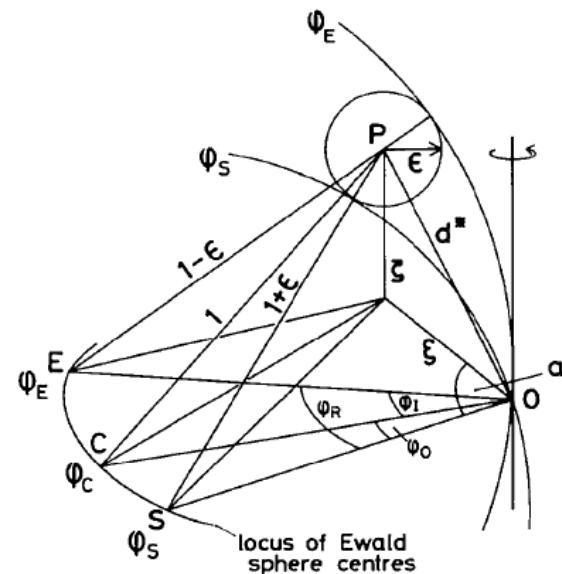
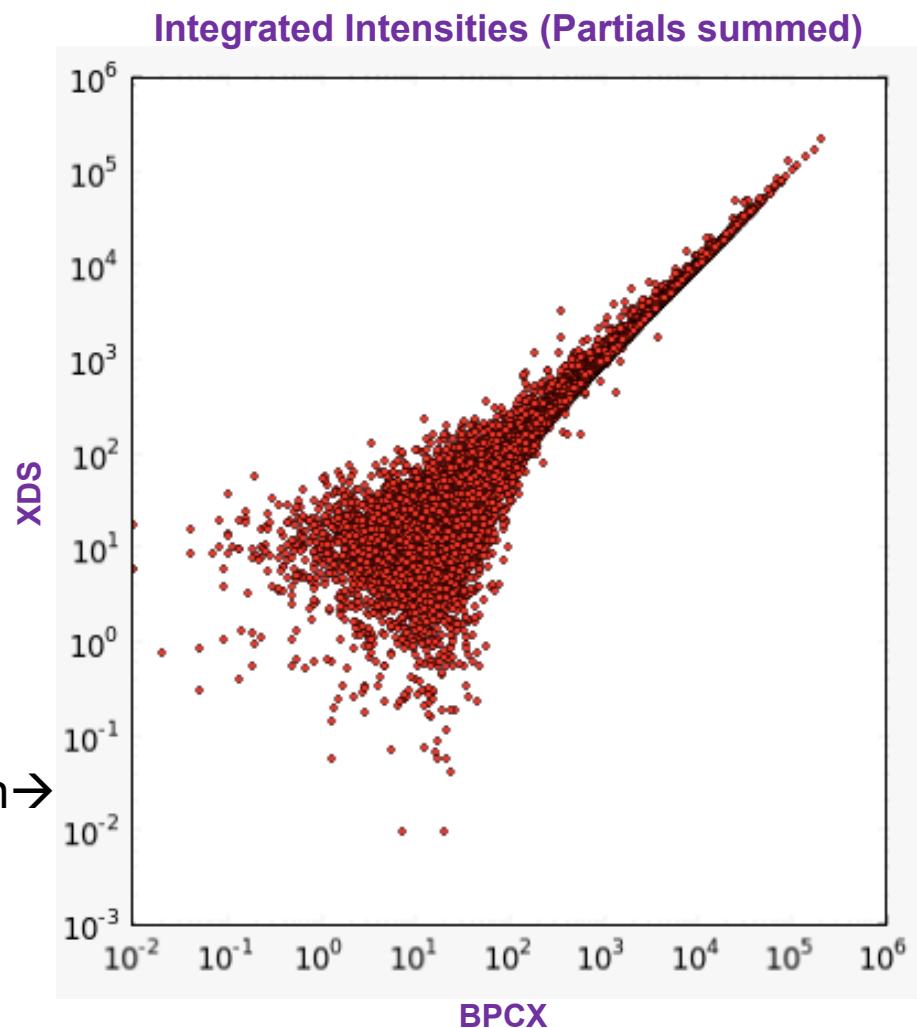


Fig. 1. The reflecting range φ_R of a reciprocal-lattice point P represented as a spherical volume, radius ε , interacting with a single geometric Ewald sphere of unit radius.

Direct comparison with XDS

- Integrate 450 frames (90°) rotation data from thaumatin ($P\ 4_12_12$) @ 1.2Å
 - 0.2° phi frames; $\sim 0.2^\circ$ mosaicity
 - Pilatus-2M detector
- XDS: 3D-profile fitting
DIALS: simple summation
...use same xtal orientation file
- Compare intensities prior to Lorentz/polarization correction; no scaling or sym-equivalent merging
- c.c. = 99.5%
 $R_{\text{scale}} = 5.1\%$
- Current: Spots → Indexing → Positional refinement → Integration → Merging
- Next few months: 3D profiles



Acknowledgements



LBNL Physical Biosciences Division

*Paul Adams
Nat Echols
Richard Gildea
Ralf Grosse-Kunstleve
Johan Hattne
Jan Kern (PSII)
Billy Poon
Vittal Yachandra (PSII)
Junko Yano (PSII)
Peter Zwart*

Diamond Light Source and CCP4

*Gwyndaf Evans
David Waterman
Graeme Winter*

Linac Coherent Light Source and SLAC

*Uwe Bergmann
Mike Bogan
Sebastien Boutet
Ashley Deacon
Philip Hart
Chris Kinney
Marc Messerschmidt
Roberto Alonso Mori
Mike Soltis
Garth Williams*

NIH / NIGMS

*Labelit: R01-GM077071
New Horizons: R01-GM095887
Phenix: P01-GM063210*

DOE

XFEL Processing: DE-AC02-05CH11231