The DIALS framework for integration software

David G. Waterman[a*], Graeme Winter[b], James M. Parkhurst[b], Luis Fuentes-Montero[b], Johan Hattne[c], Aaron Brewster[c], Nicholas K. Sauter[c], Gwyndaf Evans[b*]

[a]*CCP4, Research Complex at Harwell, Rutherford Appleton Laboratory, Didcot OX11 0FA, UK*

[b]*Diamond Light Source Ltd, Harwell Science and Innovation Campus, Didcot, OX11 0DE, UK*

[c]*Lawrence Berkeley National Laboratory, 1 Cyclotron Rd., Berkeley CA 94720, USA*

**Correspondence email addresses: david.waterman@stfc.ac.uk & gwyndaf.evans@diamond.ac.uk*

Introduction

The field of macromolecular crystallography (MX) has benefited greatly from technological advances in recent years. Automation, high brilliance beamlines at 3rd generation synchrotron sources and high frame-rate pixel array detectors have together enabled extremely rapid collection of most MX datasets. In tandem, large area photon-counting detection, microfocus beams and high quality X-ray optics have brought more challenging projects within reach. Thus there is a clear need for diffraction data analysis software designed to cope with the ever increasing volumes and rates of data collection, and with the developments in experimental methodology, from shutterless, fine-sliced rotation scans through to the thousands of randomly-oriented snapshots of serial crystallography. To match these technological advances it is to be expected that this new software would utilise techniques of parallel processing using multiple CPU and GPU machines, facilitating not just speed, but highly accurate analysis based on a comprehensive underlying physical model. Moreover the current state of the art now does not constitute the peak of progress in this field as there are significant changes yet to come, and modern diffraction integration software must be designed such that flexibility, extensibility and collaboration remain the core principles of the project.

To address the outlined requirements, we are developing DIALS (Diffraction Integration for Advanced Light Sources), a new software project for the analysis of crystallographic diffraction images. The main design goals of DIALS are versatility and modularity. To this end, the various components of a complete package have been identified and are being developed as independent modules where possible, which communicate *via* carefully designed APIs. This reduces bottlenecks caused by design decisions within one module affecting development in others. It also facilitates careful testing and benchmarking of the individual steps characteristic of integration programs, e.g. spot finding, centroid determination, indexing, orientation and geometry refinement, profile determination, profile fitting, integration etc. Thus future developments can clearly be assessed in terms of their impact on final data and structure quality. Furthermore the *toolkit* nature of DIALS will broaden its applicability in the long term beyond rotation method MX and serial MX, allowing excursions into Laue crystallography, neutron crystallography, small molecule crystallography and potentially even fibre diffraction.

At the broadest level, the component modules include, but are not limited to, visualisation of results, visualisation of diffraction images and graphical user interfaces, simulation software, databases for handling of large and multiple datasets, key algorithms for integration with specialisations for pixel-array detectors and for challenging data, and a core framework with an internal description of a diffraction experiment, into which various algorithm modules can be plugged. The EU project BioStruct-X (www.biostruct-x.org) has funded development to address these tasks and produce software that can unify the analysis of both synchrotron and FEL crystallography. Groups from Diamond Light Source, CCP4, the Center for Free Electron Lasers (CFEL), the European XFEL, EMBL Grenoble, the Paul

Scherrer Institute, Dectris and the Lawrence Berkeley National Laboratory in the USA are all contributing to aspects of the final deliverables of this project.

In this newsletter, we focus on one of those tasks, namely the development of the core programming framework with which a diffraction experiment can be described to the computer program. Formally marking this out, rather than letting it develop implicitly as part of an application suited to a particular task, is an important feature of our approach. It avoids the the core modules becoming overly application-oriented. Indeed, the framework is designed to be general with respect to hardware, to diffraction geometry and to choice of experiment, with the same basic elements shared for e.g. rotation method and serial femtosecond crystallography experiments.

Elements of the core programming framework

The framework consists of several elements - models for experimental components, cleanly defined interfaces for analysis steps, and data structures for the storage of analysis results, including reflection shoeboxes, background models and spot profiles. Where possible we are adopting standard experimental descriptions and file formats, for example imgCIF[1] and the model of experimental geometry contained therein, and HDF5[2] for the storage of results and processed data. Figure 1 illustrates how the elements of the framework can be arranged in a workflow for diffraction image integration.

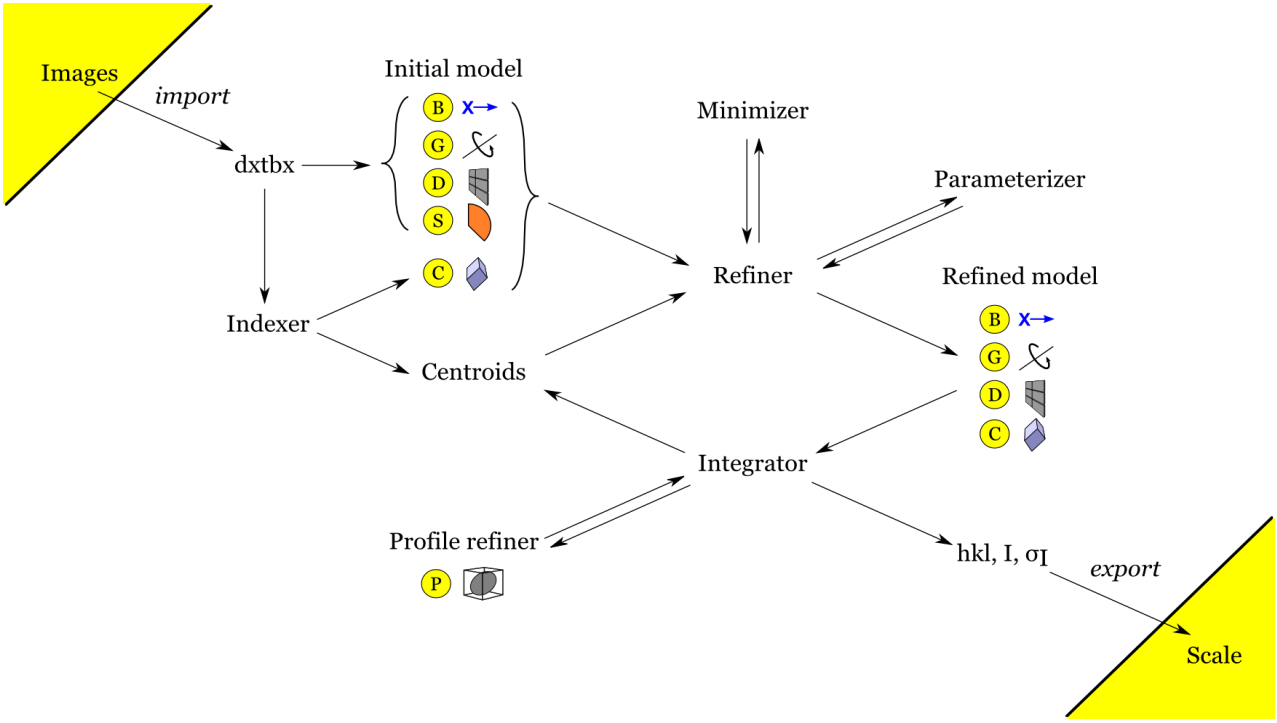

Figure 1: A possible use of DIALS framework modules to integrate diffraction images from a rotation method scan. Initial experimental models for the Beam, Goniometer, Detector and Scan (a contiguous series of rotation images) are constructed by dxtbx by analysis of the image headers and converted to the standard imgCIF frame. An indexer module accesses image data using the dxtbx and performs spot finding and indexing on the images, resulting in an initial Crystal model and calculated spot centroids. The models and centroids are passed to a refinement module. This makes a suitable parameterisation of the models, target function and choice of minimisation engine. After refinement, an improved model of the experimental geometry is passed to an integrator module. This performs spot integration using a suitable 2D or 3D method and includes a Profile forming and refinement module, which could use either empirical learning or *ab initio* synthetic methods to create the model profiles. The construction of spot profiles provides the opportunity to improve on the initial centroids, which may be used again in a second cycle through the refinement procedure. The final basic result from the integrator module is a list of reflections, their intensities and estimated intensity errors, however it is likely that additional information, such as the profile model used during integration, will also be output for potential downstream analysis.

A key objective is to model the analysis in an object-oriented manner, with a clear separation between e.g. the detector description from the sample model. As such there are four main experimental model components: the beam, the detector, the goniometer, and the sample. Each of these has a strictly defined abstract interface. For example, the detector interacts with other elements *via* abstract detector planes, accommodating any area detector that can reasonably be described by an arrangement of two-dimensional panels. Behind this abstract interface more detailed calculations may be performed, for example mapping the millimetre position on the detector surface to the appropriate pixel in the image, perhaps allowing for detector parallax in pixel array detectors or taper distortions in CCDs. Initial values for these models are generated from the headers of the raw image data from the diffraction experiment toolbox, dxtbx.

The dxtbx[3] is an extensible toolkit for providing universal access to X-ray diffraction data from a wide variety of detectors, sources and beamlines, embedding local knowledge to return a standard description of the experiment, corresponding to the imgCIF experimental geometry. In addition this toolbox also provides universal access to the three-dimensional raw pixel data to simplify access to the raw measurements and ensure that the minimum of effort is expended on straightforward elements of the implementation of a data processing toolbox. Finally the system is extensible by users and beamline scientists, such that local knowledge of how image headers should be interpreted can be added and used automatically as a "plug-in".

Collaboration and openness

We have a mandate for the free sharing of our source code, documentation and associated materials with the general public. In the spirit of the open source movement, we welcome collaboration from anyone who wishes to contribute to the project. The framework development is currently a joint effort between developers funded by BioStruct-X, Diamond and CCP4, based in the UK, and developers funded by an NIH technology project: *Realizing new horizons in X-ray crystallography data processing*, based at the LBNL, USA. Our common interests allow us to distribute effort and combine expertise effectively, greatly enhancing our joint productivity and allowing us to tackle the sizeable task of writing a new data processing package within a reasonable time frame.

To avoid "reinventing the wheel" as much as possible, the DIALS project builds on knowledge accumulated over many decades in the field of data processing for MX. We benefit greatly from the altruism of experts who contribute their ideas and advice either directly or *via* their detailed publications on existing algorithms and packages. At the heart of the DIALS framework lies a design philosophy of hardware abstraction and a generalised model of the experiment that is inspired directly by material published on the seminal workshops on position sensitive detector software[4]. Continuing in the spirit of these workshops we hold regular meetings, with talks from invited speakers, and code camps in which specific problems are addressed by intensive effort across the collaboration. Summaries of these meetings and copies of slides given as presentations are available online at http://cci.lbl.gov/dials/.

Our decision to build on existing ideas provides an obvious way to check that our developments proceed in a correct manner, as we can compare results with those from existing software. The early development of the framework code began with a series of use cases bootstrapping from data files created by processing with current software and ensuring our code reproduced acceptably similar results. We are continuing this theme by reproducing published spot finding and integration algorithms used by integration packages such as XDS[5] and MOSFLM[6] and from software used for other types of diffraction image analysis, such as ESMERALDA[7]. Nevertheless, the toolkit architecture implies that we are not limited by these initial methods. We expect new research to lead to more advanced or case-

specific algorithms in future, which may either be implemented by us or by other groups who chose to work with the toolkit.

Source code and timescales

The DIALS framework is being developed in a fully open-source, collaborative environment. We are using Python plus C++, with heavy use of the cctbx[8] for core crystallographic calculations and much infrastructure including a complete build system. Seamless interaction between the C++ and Python components of this *hybrid system* is enabled by boost.python. Python provides a useful ground for rapid prototyping, after which core algorithms and data structures may be transferred over to C++ for speed. High level interfaces of the hybrid system remain in Python, facilitating further development and code reuse both within DIALS and by third parties.

The DIALS framework has a home on sourceforge http://dials.sourceforge.net/. A beta release of DIALS is planned for late 2014 with a final release following in 2015. This complete diffraction integration package will be distributed by CCP4.

Acknowledgements

JMP and LF-M were supported by the European Community's Seventh Framework Programme (FP7/2007-2013) under BioStruct-X (grant agreement N°283570). N.K.S., A.B., and J.H. were supported by National Institutes of Health/National Institute of General Medical Sciences grants 1R01GM095887 and 1R01GM102520, as well as by the Director, Office of Science, Department of Energy under Contract DE-AC02-05CH11231.

References

[1] Bernstein, H. J. & Hammersley, A. P. (2005). *International Tables for Crystallography*, Vol. G, *Definition and Exchange of Crystallographic Data*, 37-43.

[2] HDF Group (2010). *Hierarchical Data Format Version 5*, http://www.hdfgroup.org/HDF5.

[3] Parkhurst *et al*. (2013), in preparation.

[4] Bricogne, G. (1987). *Proceedings of the CCP4 Daresbury Study Weekend*, pp. 120-145.

[5] Kabsch, W. (2010). *Acta Cryst.* D**66**, 125-132.

[6] Leslie, A. G. W. and Powell H. R. (2007), *Evolving Methods for Macromolecular Crystallography*, **245**, 41-51. ISBN 978-1-4020-6314-5.

[7] Fuentes-Montero, L., Cermak, P. & Rodriguez-Carvajal, J., http://lauesuite.com, in preparation.

[8] Grosse-Kunstleve, R. W., Sauter, N. K., Moriarty, N. W., & Adams, P. D. (2002). *Journal of Applied Crystallography*. **35**, 126–136.