

AUTOMATION OF THE MACROMOLECULAR CRYSTALLOGRAPHY BEAMLINES AT THE CANADIAN LIGHT SOURCE

M.N. Fodje*, R. Berg, G. Black, P. Grochulski, K. Janzen, Canadian Light Source, 101 Perimeter Road, Saskatoon, SK, Canada S7N 0XN.

Abstract

The Canadian Macromolecular Crystallography Facility (CMCF) is a suite of two beamlines 08ID-1 and 08B1-1. Beamline 08ID-1, is an undulator beamline for studying small crystals and crystals with large unit cells, while beamline 08B1-1 is a bending-magnet beamline for high-throughput macromolecular crystallography with a high level of automation. The primary method of access to CMCF 08B1-1 will be remote, in what is commonly referred to in the field as "Mail-in" crystallography. We are developing a software system for automating both beamlines, with modules for beamline control, experiment control, data analysis, information management, and graphical user interaction. The system is developed using the Python programming language and makes use of popular open-source frameworks such as Twisted, Django and GTK+. Once completed, the system will allow automation of the macromolecular crystallography experiment from experiment setup to data analysis, thereby increasing the efficiency of the CMCF beamlines and reducing the need for user travel to the synchrotron.

BACKGROUND

The growing impact of macromolecular structural analysis to pharmaceutical, academic and industrial research has resulted in a growing demand for access to protein crystallography beamlines. This demand is reflected not only in the number of samples available for analysis, but also in the increased number of scientists from different fields now using structural information in their research. As a result, many more users with less crystallographic training are demanding access to macromolecular crystallography (MX) beamlines at synchrotron facilities. Fortunately, the MX experiment is highly amenable to automation [1]. It is not surprising therefore that there are many on-going efforts by various synchrotron facilities to provide highly automated MX beamlines to the community of users [1-3].

The synchrotron MX experiment can be broken down into distinct steps (see Table. 1). These include sample preparation, beamline setup, sample mounting, sample alignment, sample characterisation, data acquisition and data processing. The details of each step may vary based on the specific sample being examined and type of experiment desired. With the exception of the first step, which is usually carried out by experimenters at their home laboratories, the remaining steps can be automated to a very high degree. It is therefore possible in principle to build a fully automated beamline where experimenters simply prepare and send their samples to the beamline,

data is automatically acquired, and experimenters are never needed on-site.

Table 1: Steps involved in an MX experiment

Step	Description
Sample Preparation	Samples are frozen in cryogen at the home laboratory and couriered to the synchrotron by experimenters
Beamline Setup	The beamline is configured and optimized
Sample Mounting	The sample is mounted on a Goniometer
Sample Alignment	The sample on the Goniometer is positioned such that the sample rotates within the X-ray beam, for data acquisition
Sample Characterisation	Initial data frames are collected and processed to obtain improved parameters to be used for data acquisition
Data Acquisition	Data frames are collected
Data processing	Data frames are integrated and reduced to reflection files for further analysis and structure determination by the users

Automation of an MX beamline requires tight integration of various hardware and software components. In addition to the beamline hardware required for delivery of a high-quality and stable beam at the sample position, robotic sample mounting devices and computer hardware for data processing are also required. The software system is a central component of every automated beamline and great care has to be taken to ensure that it is reliable and enables the acquisition of the best possible data. Here, we describe the architecture and implementation of the software for automation of the 08B1-1 and eventually the 08ID-1 beamlines at the Canadian Light Source (CLS).

SOFTWARE ARCHITECTURE

The software system being developed for automation of the CMCF Beamlines is a modular system, layered above the low-level beamline instrument control system which is based on the Experimental Physics and Industrial Control System (EPICS). The main system modules are the Experiment Management Module (EMM), the Beamline Control Module (BCM), the Data Processing Module (DPM), and the Information Management Module (LIMS) (see Fig. 1).

* Email: Michel.fodje@lightsource.ca

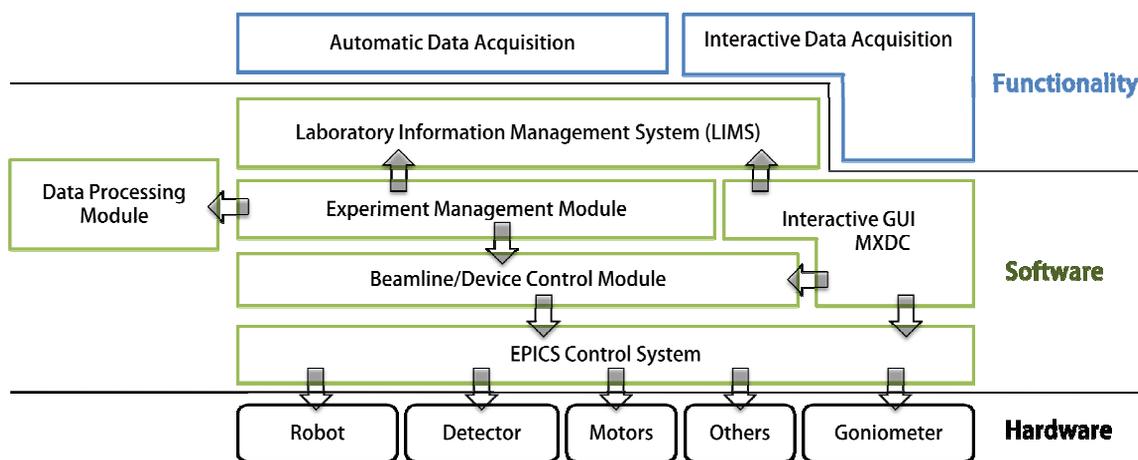


Figure 1: CMCF Software Architecture for automation, showing the various software modules, cross communication between the modules, and provided functionality.

Information Management Module

The Information Management Module, also known as the Laboratory Information Management System (LIMS) is responsible for the storage of information about samples, sample shipments, experiment requests, experiment results and data sets. This module provides a web-based interface for users to submit sample information and review experiment results, and also a web-based interface for beamline staff to manage beamline sessions. It also provides crystal information and experimental requests to the EMM.

Beamline Control Module

The Beamline Control Module (BCM) is responsible for directly controlling the beamline hardware. This module is a high-level module, which must be distinguished from the EPICS based beamline control system. The BCM is an integrated unit which controls the beamline hardware through the EPICS Channel Access protocol, in order to carry out the following functions automatically:

- Beamline configuration.
- Beam optimization.
- Sample mounting and dismounting.
- Sample alignment.
- Data acquisition.

Data Processing Module

The Data Processing Module (DPM) is in charge of integrating and reducing collected data frames into crystallographic reflection files. Specifically the DPM carries out the following functions:

- Scoring of samples to assess quality and suitability for data acquisition.
- Determination of sample parameters and an optimum strategy for data acquisition.

- Integration and reduction of diffraction images into reflection files.
- Data conversion into user-friendly formats for further processing.

Experiment Management Module

The experiment management module (EMM) is at the top of the of the beamline software hierarchy. The role of this module is to substitute the experimenter in carrying out scheduling and coordination of the steps involved in the MX experiment. As a result, the functioning of this module is experiment-centric rather than beamline-centric. This module delegates tasks to the other modules based on experiment information received from the LIMS. For example, sample mounting is delegated to the BCM, while sample characterisation is delegated to the DPM.

IMPLEMENTATION

The beamline software system is implemented in the Python programming language. Python is an interpreted, object oriented, high-level programming language, suited for rapid application development [4]. The availability of several high-quality Python-based frameworks for graphical interface development, web-application development and network server-client development made Python an obvious choice for the implementation language.

The LIMS is implemented using the Django web-application framework [5]. External interfaces used for exchanging information with other modules such as the Experiment Management Module, are implemented as JSON-RPC (JavaScript Object Notation Remote Procedure Call) interfaces [6].

The other modules, the BCM, EMM and DPM, are implemented using a combination of the Twisted Framework, and the Glib/GObject system. The Glib/GObject system, a part of the GTK+ toolkit [7] is a low level library which provides interfaces for event-

driven programming together with a dynamic object system. It is widely used for graphical interface development but can be used for non-graphical applications. The BCM uses GObject extensively to achieve an event driven architecture, where changes in the control system are propagated asynchronously to the rest of the system as they occur. For example, data acquisition is automatically paused when the synchrotron beam is no longer available. The GTK+ toolkit is also used for all interactive graphical interfaces such as the MX Data Collector (MXDC) (used for interactive data acquisition at the beamlines).

The Twisted Framework is a highly flexible, secure and stable networking engine written in Python with interfaces for inter-process communication, asynchronous programming, and web-application development [8]. The Perspective Broker modules are used for all communication between the BCM, DPM and EMM. This includes remote method calls and transmission of data objects. Twisted is also used to provide remote administrative python consoles for live debugging of the modules.

The BCM, DPM and EMM also make use of the Multicast DNS (Domain Name System) service discovery protocols to publish and discover configuration parameters of available services to which other modules may connect. For example, the EMM automatically determines at run-time the host address and port of the BCM, and DPM services which can be running on any machine within the local network. In addition, services can easily be migrated to different hosts and other modules will automatically be notified to reconnect at run-time without manual reconfiguration.

The software system relies on existing established software packages from the crystallographic community for specialized functions. Specifically, the BCM makes use of the XREC package [9] for automatic alignment of samples. Furthermore, the DPM makes use of the XDS [10], CCP4 [11] and BEST [12] software packages.

CONCLUSION

The CMCF Software system will enable remote access to the facility in what is usually referred to as "Mail-in" crystallography, with a high level of automation. Users will prepare their samples at their home labs and ship the samples to the CLS. Using information provided by the users through the web-based LIMS, data will be automatically collected by the EMM delegating to the BCM and DPM as appropriate. After review of the results by beamline staff, the results will be made available to users through the LIMS. When fully functional, this mode of operation will increase the efficiency of the beamlines and ultimately the number of samples that can be analyzed.

ACKNOWLEDGEMENT

The Canadian Light Source is funded by the University of Saskatchewan, the Canadian Institutes of Health Research (CIHR), the National Research Council (NRC), the Natural Sciences and Engineering Research Council (NSERC), the government of Saskatchewan, and Western Economic Diversification Canada

REFERENCES

- [1] S. Arzt, *et al.* "Automation of macromolecular crystallography beamlines," *Progress in Biophysics and Molecular Biology* 89 (2005) p124–152.
- [2] S.M. Soltis *et al.* "New paradigm for macromolecular crystallography experiments at SSRL: automated crystal screening and remote data collection," *Acta Cryst.* (2008). D64, p1210-1221.
- [3] G. Snell, *et al.* "Automated Sample Mounting and Alignment System for Biological Crystallography at a Synchrotron Source," *Structure*, 12 (2004), p537–545.
- [4] G. van Rossum. <http://www.python.org> ,
- [5] Django Project. <http://www.djangoproject.org> .
- [6] JSON-RPC Working Group. "JSON-RPC 1.1 Specification", <http://www.json-rpc.org>.
- [7] GTK+ Toolkit, <http://www.gtk.org>.
- [8] The Twisted Framework, <http://twistedmatrix.com>.
- [9] S.B. Pothineni, T. Strutz and V.S. Lamzin. "Automated detection and centring of cryo-collid protein crystals," *Acta Cryst.* D62 (2006), p1358-1368.
- [10] Collaborative Computational Project, Number 4. "The CCP4 Suite: Programs for Protein Crystallography," *Acta Cryst.* D50 (1994), p760-763.
- [11] Collaborative Computational Project, Number 4. "The CCP4 Suite: Programs for Protein Crystallography," *Acta Cryst.* D50 (1994), p760-763.
- [12] G.P. Bourenkov and A.N. Popov "A quantitative approach to data-collection strategies" *Acta Cryst.* D62 (2006), p58-64.