An overview of the CCP4 project in protein crystallography: an example of a collaborative project

M. D. Winn

Daresbury Laboratory, Daresbury, Warrington WA4 4AD, UK. E-mail: m.d.winn@dl.ac.uk

The Collaborative Computational Project Number 4 (CCP4) was established in 1979 to promote collaboration between UK groups writing software for protein crystallography. CCP4 now distributes a large software suite and is active in developing new software. Equally importantly, CCP4 provides a focus for the whole protein crystallography community *via* meetings, workshops, email lists and various publications. In this article, an overview is given of CCP4 activities and their administration. The emphasis is on generic features of the collaboration rather than details specific to protein crystallography. The CCP4 model has inspired similar developments in NMR, and it is hoped that the biological XAS community may pursue similar collaboration.

Keywords: collaboration; software; protein crystallography.

1. Introduction

The Collaborative Computational Project Number 4 (CCP4) was established in 1979 by the UK Science and Engineering Research Council (SERC) to promote collaboration between UK groups writing software for protein crystallography (PX) (Collaborative Computational Project, Number 4, 1994; Dodson *et al.*, 1997). With continued support from SERC and the Biotechnology and Biological Sciences Research Council (BBSRC), CCP4 has grown in size and influence. Its aims are to develop, maintain and distribute state-of-the-art software for PX and to promote collaboration and education within the whole PX community. In the following sections, CCP4's activities in software development and education will be summarized. Details of all CCP4 activities can be found *via* the web site www.ccp4.ac.uk.

2. CCP4 software suite

The CCP4 software suite currently exists as around 160 separate programs covering most aspects of protein crystallography from initial data processing, through phasing and model building, to refinement and validation of the final structure. A protein structure can be solved using a subset of these programs, with the choice of programs being determined by the nature of the particular experiment and any problems encountered. There is, moreover, some redundancy in the programs available and hence some flexibility in how the suite is used.

The form of the suite as a loose collection of programs has arisen from the collaborative nature of CCP4, with programs coming from a variety of sources. Traditionally, most programs have been written in Fortran, with a few additional C and Java applications. In future, there are likely to be an increasing number of C, C++ and Python applications. In principle, a program in any language can be included in the suite with very few changes. Therefore, state-of-the-art software can be made available to biologists within a short timescale and with minimal burden on the developer.

Nevertheless, it is helpful to the developer and the user to have a standardized framework in which to work. At a basic level, this framework is supported by a software library that covers standard operations in crystallographic computing, for example, platform-dependent file i/o, application of crystallographic symmetry operators, fast Fourier transforms, provision of atomic form factors *etc.* Developers may use this library to simplify production of code. In addition, use of the library encourages a common look-and-feel of programs.

When using several programs to solve a structure, data is passed between programs using standard file formats. The following standard file formats are currently supported:

Reflection data: MTZ file, consisting of a reflection list + header information.

Map and mask format: CCP4 map format, consisting of a three-dimensional array + header information.

Coordinate formats: PDB format (only the subset of record types needed for a working coordinate format are used by the programs).

mmCIF format: the macromolecular crystallographic information file is used for data harvesting and for geometric restraint information.

Library routines are provided to read and write these file formats. There are also a number of utilities to convert to and from external file formats.

On the question of formats, the trend today is towards well defined abstract data models, from which specific implementations can be derived. In this context, file formats act as particular views of that data model. CCP4 is currently developing data models for reflection, map and coordinate data, with file formats playing a less central role. These data models are implemented in C structures and C++ classes. The CCPN project (Collaborative Computing Project for NMR, 2001) is following a more formal approach, using the Unified Modelling Language (UML) to describe a data model.

There is a demand today for user-friendly graphical interfaces. This is certainly true in protein crystallography, where users are increasingly likely to have a biological background rather than a crystallographic one and are more likely to be familiar with a Windows environment than a UNIX one. CCP4 has therefore developed a Tcl/Tk-based Graphical User Interface (GUI) for the suite called *ccp*4*i*.

The aim was to have an interface that is easy to use and has the same look-and-feel across the suite but which nevertheless does not place any demands on the underlying programs. The interface was therefore designed as a separate layer that prepares input scripts for the programs. Thus, the user may still run the programs directly if desired and the developer is able to work on the program to some extent free from the interface.

In *ccp*4*i*, programs are presented as 'tasks', each designed to do a particular job in the crystallographic process. The task may involve a sequence of CCP4 programs. In particular, so-called 'jiffy' programs for performing simple jobs such as format conversion are included seamlessly. The individual task interfaces are maintained either by the appropriate program author or by CCP4 staff.

As well as preparing the input, *ccp*4*i* provides easy access to tools for viewing the program output, an integrated help system and a simple database facility for managing projects. The latter presents a list of jobs run, and selecting a job allows the user to review the input and output and perhaps to rerun the job with a slight change to the input.

The CCP4 suite of programs, together with *ccp*4*i*, documentation and examples, is distributed as well defined releases. The latest version, 4.2, was released in April 2002. The provision of well defined

releases is convenient for the end user and ensures compatibility between the constituent programs of the suite. The distribution includes source code for all programs, as well as binaries for the most common platforms (SGI, alpha, Linux PCs, Windows NT and now Mac OS X). The suite is expected to compile and run on most UNIX-based systems, as well as Windows NT and Windows 2000, and standard procedures are included for compiling the source code on all commonly occurring platforms.

All users of the CCP4 suite must sign a licence agreement. The licence distinguishes two main classes of software. Part (i) software (consisting of the software library and other core code) can be copied and modified with few restrictions, and the aim is that this software be widely used. For Part (ii) software (consisting of the application programs), the conditions of the original authors are retained. The licence is free of cost to non-profit-making organizations, but an annual licence fee is charged to commercial organizations.

In the post-genomic era, correct and complete archiving of structural data is very important. Enough information should be included to allow future validation of the data and to allow for possible reinterpretation. The Data Harvesting initiative promoted by the Macromolecular Structure Database group at the European Bioinformatics Institute aims to capture information from the structure-solution process at the time it is created. For crystallographically determined data, CCP4 includes Data Harvesting software for recording information about data processing, phasing and structure refinement (Winn, 1999). This information is uploaded to the Protein Data Bank at the time of structure deposition. Similar initiatives are underway for NMR and electron microscopy.

The most recent initiative of CCP4 is the development of a molecular graphics program that will be used for the solution and analysis of macromolecular structures (Potterton, 2001). Early implementations will allow the viewing and manipulation of macromolecular structures, while eventually the molecular graphics viewer will act as a gateway to most of the CCP4 software. Other areas of interest to CCP4 are increasing automation of the structure solution process, integration with data collection software and increasing links with other areas of structural biology.

3. CCP4 educational activities

An important part of CCP4's remit is concerned with education. The CCP4 group sponsors and/or organizes a wide range of educational activities, and these are intended to cover all of protein crystallography and are by no means restricted to the CCP4 software suite. The flagship event is undoubtedly the annual study weekend held at the beginning of January. Each year, an area of protein crystallography is chosen as the topic. Speakers cover background material on the topic suitable for students as well as recent developments in the field. Recent study weekends have been:

2002 York, *High-Throughput Structure Determination*;
2001 York, *Molecular Replacement and its Relatives*;
2000 York, *Low Resolution Phasing*;
1999 Sheffield, *Data Collection and Processing*;
1998 Reading, *Databases for Macromolecular Crystallographers.*

The proceedings of these five study weekends have been published as special issues of *Acta Crystallographica Section D* (Proceedings of the CCP4 Study Weekend, 1998, 1999, 2000, 2001, 2002). The proceedings of previous study weekends (going back to 1980) were published by CLRC Daresbury Laboratory.

CCP4 runs other workshops on an occasional basis. These are often of a practical nature, such as the week-long workshop on macromolecular refinement held in York in January 2001, where 40 students were taught *via* a mix of lectures and computer practicals. A one-day introduction to the CCP4 suite has been given several times, for example, as a workshop of the IUCr Congress in Glasgow, 1999. A number of seminars on aspects of the CCP4 suite have been held at Daresbury Laboratory, for example, Phil Evans on data processing with the program *scala*. The latter talk was recorded and is available on the CCP4 web site. Finally, CCP4 supports several meetings financially, such as the annual UK Summer School in Protein Crystallography.

The CCP4 group also supports education *via* the production of literature and *via* electronic services. In addition to the proceedings of the study weekends, CCP4 also produces a newsletter approximately every six months. The newsletter contains news of CCP4 events and software developments and contributed articles in the general area of PX software development and usage. Although the CCP4 newsletter is not a formal publication, some articles have become standard references in the primary literature.

The CCP4 web site (http://www.ccp4.ac.uk/) is designed to be the first port of call to find out about CCP4 activities. There are links to news items, software documentation, information on downloading software, problems pages, information on development projects, details of courses *etc.* CCP4 runs several email distribution lists; for example 'ccp4bb' (a.k.a. the 'Bulletin Board') is a community-wide list for general protein crystallography, and 'ccp4-dev' is a general list for software developers. Other lists cater for specific projects or committees. Finally, the help desk address ccp4@ccp4.ac.uk provides direct user support.

4. Organization

CCP4 is a UK collaboration supported by the UK Biotechnology and Biological Sciences Research Council (BBSRC). As such, it aims to represent the UK PX community, who have input *via* two 'working groups'. Working Group 1, largely comprising heads of groups, meets annually and makes general policy. Working Group 2 meets three times a year and oversees the organization of the study weekend and the development of the software suite. A smaller executive committee, drawn from the two working groups, makes operational decisions based on agreed policy. The various activities of CCP4 are coordinated by the CCP4 group at Daresbury, who also have responsibility for the production of the public releases of the software suite.

CCP4 employs a number of staff directly, funded by the BBSRC and by commercial income. These staff provide for the core activities of the collaboration, development of the CCP4 software library and development of key programs. However, the contributions of those not funded by CCP4 are extremely important and historically were vital to the growth of CCP4. These collaborators donate software to the suite, provide advice on the development of the core code and contribute to the educational workshops. It should be stressed that many of these collaborators are based outside the UK and provide an international dimension to CCP4.

Software is accepted into the CCP4 suite at the discretion of the CCP4 group at Daresbury. The donated software must obviously provide useful functionality. Overlap with existing functionality is not necessarily a problem, since implementations are rarely identical and a specific implementation may be advantageous in certain cases. Authors are strongly encouraged to utilize CCP4 software libraries, to ease maintainability and portability and to provide a CCP4 look-and-feel, but this is not a strict requirement. Finally, source code must be provided, as this is considered an integral part of the distribution. The author of donated software receives no financial reward. However,

the author is able to tap into an established user base and thereby obtain an increased profile in the biological community as well as valuable feedback.

5. Conclusions

In the previous sections, the current status and activities of CCP4 have been outlined. CCP4 is generally considered to be one of the most successful of the UK's CCP projects and was, for example, used as a model for the CCPN collaborative project for the NMR community [Collaborative Computing Project for NMR (2001); see also Collaborative Computational Project, Number 11 (2001) for another example of a successful CCP in a biological area]. In the context of this special issue for BioXAS, it is pertinent to consider what factors have made CCP4 successful and whether these can be applied to BioXAS.

The CCP4 software suite has now reached a critical mass whereby it is advantageous for software developers to collaborate to some extent with CCP4. There are over 500 registered user sites worldwide, and, while CCP4 is rarely used exclusively at these sites, developers can assume the existence of the CCP4 environment at a majority of protein crystallography laboratories. Where software is maintained independently, programs are often still based on CCP4 data formats and protocols. This situation has been reached by steady growth over two decades. The two factors that have driven this growth are the development of core library routines and the willingness of software developers to contribute to the project. The former eases the burden of writing software, while ensuring a commonality that is of major benefit to the end user. The latter factor is more intangible but is crucial. More can be achieved as a community if developers pool their resources. Developers also benefit from the increase in feedback obtained, both from collaborating developers and from the established pool of users. Clearly, there is conflict here with the growing efforts of institutions to commercialize the products of their employees, and this issue would need to be addressed for a new project.

From the point of view of the biologist, it is important to maintain a non-commercial option for software. This software also needs to be accessible to the biologist who may not have been trained in the underlying theory. In CCP4, this has been achieved through the development of the GUI, the provision of extensive documentation and the support of an active user community. The latter is fostered by the regular CCP4 meetings, by the 'Bulletin Board' and by the other educational services, as described above. It is important not to underestimate the effort involved in running these services. An email distribution list is easy to set up, but to be successful requires attention to technical aspects (subscription, filtering of spam *etc.*) as well as active participation by members of the field. Organization of conferences, workshops and meetings is of course time-consuming, particularly the preparation of coursework and computer practicals. The latter, however, are frequently the most useful resources for students new to the field.

The activities of CCP4, both in software development and in education, require a significant amount of administration. This has traditionally been provided by the programmers at Daresbury Laboratory, although in recent years an administrative assistant has been employed to deal with much of the non-scientific work. A new collaborative project is likely to have more modest aims (as CCP4 did initially). Nevertheless, success of a large-scale collaboration does depend on efficient administration, with a clearly identified point-of-contact that responds quickly to problems.

MDW is supported by the Biotechnology and Biological Sciences Research Council through the CCP4 grant (B10200). CCP4 has benefited from the activities of many people past and present, including staff employed on the project at Daresbury Laboratory and UK universities, contributors of software, collaborating developers, and users of the software who have given feedback. As a community-wide collaboration, it is impossible to name all contributors, but their contributions are gratefully acknowledged.

References

Collaborative Computational Project, Number 4 (1994). *Acta Cryst.* D**50**, 760–763.
Collaborative Computational Project, Number 11 (2001). *TBR – The Bioinformatics Resource*, http://www.hgmp.mrc.ac.uk/CCP11/index.jsp.
Collaborative Computing Project for NMR (2001). *A Collaborative Computing Project for the NMR Community*, http://www.bio.cam.ac.uk/nmr/ccp/.
Dodson, E. J., Winn, M. D. & Ralph, A. C. (1997). *Methods Enzymol.* **277**, 620–633.
Potterton, E. (2001). *CCP4 Newsletter*, No. 39, Article 6.
Proceedings of the CCP4 Study Weekend (1998). *Acta Cryst.* D**54**, 1065–1206.
Proceedings of the CCP4 Study Weekend (1999). *Acta Cryst.* D**55**, 1631–1772.
Proceedings of the CCP4 Study Weekend (2000). *Acta Cryst.* D**56**, 1205–1357.
Proceedings of the CCP4 Study Weekend (2001). *Acta Cryst.* D**57**, 1355–1490.
Proceedings of the CCP4 Study Weekend (2002). *Acta Cryst.* D**58**, 1897–1970.
Winn, M. D. (1999). *CCP4 Newsletter*, No. 37, Article 13.

 Received 29 November 2001 • Accepted 29 August 2002