MS41 Training computational method developers

Chairs: Dr. Marijana Dakovic, Dr. Martin Lutz

MS41-O1

Developing new crystallographic software: how and what to do?

Juan Rodriguez-Carvajal[1]

1. Diffraction Group, Institut Laue-Langevin, Grenoble, France

email: jrc@ill.eu

Crystallographic software is nowadays distributed and used worldwide. The incursion of commercial programs in the arena of crystallographic computing has had as a consequence that the number of crystallographers able to contribute to the development of new tools and algorithms for handling new problems seems to be diminishing. There are still many specialized areas, where crystallographic software development is still worthwhile, in which commercial companies are not too much interested due to the lack of a profitable market. New developers should be attracted to the field of open crystallographic computing and they should be aware of open-source existing crystallographic libraries in order to not reinvent the wheel. The availability of new high level languages, like Python, has facilitated a lot the possibility of making graphic user interfaces (GUI) and write quite compact useful programs. We do not have to forget that, if we want to enter into de core of the crystallographic and mathematical algorithms as well as the development of relatively large projects, the use of statically highly-typed compiled languages (like Fortran 95/2003/2008 or C/C++) is much more efficient in the long term [1]. After presenting existing open libraries, my views on scientific software development and how to proceed in crystallographic computing, in this communication I will review, as a detailed example, the current status of the library ***CrysFML*** [2] which is the base for many crystallographic programs of the ***FullProf Suite*** [3]. ***CrysFML*** is a library written in Fortran 95 containing modules with plenty of procedures for elaborating crystallographic programs. A series of high level procedures are available so that the programming of relatively complex problems can be done easily. Some examples will be described. The library is open source (LGPL licence) and it is at the web-site: https://forge.epn-campus.eu/projects/crysfml/repository. As projected plans for future improvements we intend to develop a wrapper of the library to make the high level procedures available to the C-language (and hence to Python). A future version, in Fortran 2008/2018, fully exploiting the object oriented paradigm is currently being developed.

References:

[1] Juan Rodríguez-Carvajal (2004), Compcomm Newsletter 3, 32-40 .Available at https://www.iucr.org/__data/assets/pdf_file/0008/6389/iucrcompcomm_jan2004.pdf

[2] Juan Rodríguez-Carvajal and Javier González-Platas (2003), Compcomm Newsletter 1, 50-58. Available at https://www.iucr.org/__data/assets/pdf_file/0009/6399/iucrcompcomm_jan2003.pdf

[3] Juan Rodríguez-Carvajal (1993), Physica B 192, 55-69

Keywords: scientific software, crystallographic computing, Fortran/C/C++ languages