Developing cross-platform delivery for release and development versions of the Felix suite of electron diffraction visualisation codes

The symmetry of a crystal normally determines its functional properties. This is equally true on the nano-scale as it is at the macro-scale. Whilst for bulk material the structure and symmetry can routinely be solved by X-ray diffraction, there is no comparable technique for nanostructured materials. Electron diffraction has the required nano-scale resolution and sensitivity, but overlapping data from different diffracted beams has limited its use to date. We recently demonstrated that computer control of beam tilt and image capture in a conventional transmission electron microscope (TEM) can be used to overcome this problem, quickly providing very rich diffraction datasets [1].

The method is most productive when using in conjunction with computational simulation of expected diffraction results based on the structure and symmetries of the input crystal. We have recently developed such a computer code written in Fortran 90, based on what is known as the Bloch-wave method. The code is fully parallelised for MPI-type architectures and has been shown to have linear speed-up. It consists of two parts, (1) FELIXSIM for simulating diffraction patterns from a given input crystal structure (described via standard crystal information files .cif) and (2) FELIXREFINE which uses FELIXSIM iteratively together with an minimization algorithm in order to refine a .cif file based on experimental diffraction images. Both codes are available as open source via github [2].

At the moment, the code has been released to the CSC systems and, via CSC’s open build system [3], compiled for SuSe Linux distributions. However, we would like to make the code available across as many platforms as possible. In principle, this should be easy since the code compiles using standard the gfortran compiler as well as standard numerical libraries such as BLAS, LaPack and fftw3. In practice, it will require shell scripting, designing a dedicated web page (and interfacing with github), development of a build, release and debug model, etc. Hence the project will give you a flavour of modern computer science techniques in delivering applications to market. It will be ideal for a student contemplating research/work in this area. A strong background in programming is recommended. For more information please contact Rudolf Roemer (R.Roemer@warwick.ac.uk) or see http://www.warwick.ac.uk/go/DisQS.

2. https://github.com/RudoRoemer/Felix
3. https://obs.csc.warwick.ac.uk/