Journal of
Applied Crystallography

ISSN 0021-8898

Received 6 January 2006
Accepted 28 February 2006

Mercury: visualization and analysis of crystal structures

Clare F. Macrae,* Paul R. Edgington, Patrick McCabe, Elna Pidcock, Greg P. Shields, Robin Taylor, Matthew Towler and Jacco van de Streek

Cambridge Crystallographic Data Centre, 12 Union Road, Cambridge CB2 1EZ, UK. Correspondence e-mail: macrae@ccdc.cam.ac.uk

Since its original release, the popular crystal structure visualization program *Mercury* has undergone continuous further development. Comparisons between crystal structures are facilitated by the ability to display multiple structures simultaneously and to overlay them. Improvements have been made to many aspects of the visual display, including the addition of depth cueing, and highly customizable lighting and background effects. Textual and numeric data associated with structures can be shown in tables or spreadsheets, the latter opening up new ways of interacting with the visual display. Atomic displacement ellipsoids, calculated powder diffraction patterns and predicted morphologies can now be shown. Some limited molecular-editing capabilities have been added. The object-oriented nature of the C++ libraries underlying *Mercury* makes it easy to re-use the code in other applications, and this has facilitated three-dimensional visualization in several other programs produced by the Cambridge Crystallographic Data Centre.

1. Introduction

Mercury (Fig. 1) is a crystal structure visualization program produced by the Cambridge Crystallographic Data Centre (CCDC). A description of the program was published when it was first released (Bruno *et al.*, 2002), together with a report of its application to the study of packing of mono- and di-alcohols (Taylor & Macrae, 2001). Since the first release, in 2001, the application has become extremely popular, with approximately 2800 users signed on to the *Mercury* e-mail announcement list, and, based on the number of downloads, several thousand active users of the program worldwide. This paper gives a brief overview of *Mercury*, followed by more detailed descriptions of some of the features that have been added since the original publication of 2002.

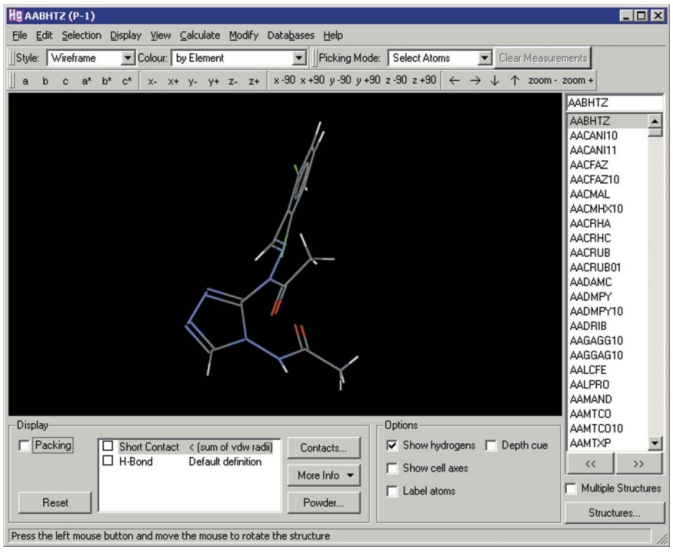


Figure 1
The main window after opening Mercury.

2. Overview

Although it has several uses, the core purpose of *Mercury* is to help scientists understand the crystal structure packing arrangements of organic and organometallic molecules. Above all, this requires good visualization and the ability to identify and show the key intermolecular interactions that hold a crystal structure together. Visualization options include the display of molecules in a variety of styles and colouring schemes; calculation and display of centroids, and least-squares and Miller planes; and the ability to switch the display of individual atoms or molecules on or off, or to show the contents of any number of unit cells (or fractions of unit cells). Items such as atom labels and unit-cell axes can optionally be displayed. The three-dimensional image can be rotated, translated and scaled by use of the mouse, a dial-box or buttons on a tool bar. Crystal structures may be viewed along direct or reciprocal cell axes, or perpendicular to any specified plane or bond. Coupled with this general display functionality is the ability to specify any type of non-bonded contact (*e.g.* a particular type of hydrogen bond) and then show all contacts of that type formed by the molecule(s) on display (Fig. 2*a*). Clicking on an individual contact causes the neighbouring molecule to be displayed in full, so that networks of molecules and their linking contacts can rapidly be constructed (Fig. 2*b*). This feature, copied from the *RPluto* program (Motherwell *et al.*, 1999), is central to the high levels of usability that *Mercury* provides. Undo and redo features facilitate interactive exploration of crystal structures, since any expansion of the molecular ensemble on display can be reversed if it does not add to the user's comprehension of the packing arrangement.

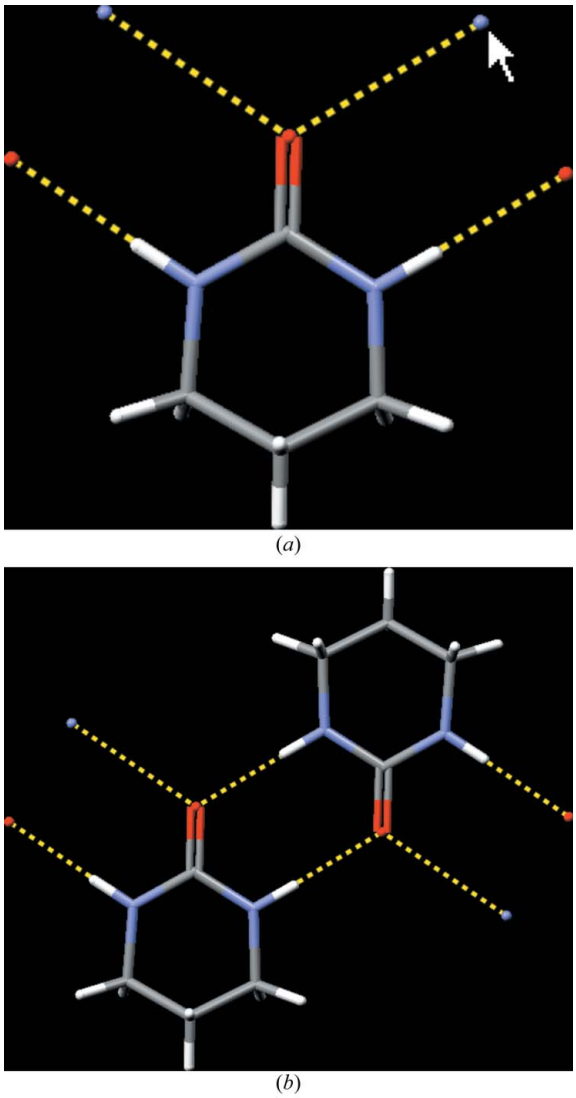


Figure 2
(*a*) The hydrogen bonds formed by a molecule in the crystal structure of perhydropyrimidin-2-one (CSD refcode APYFEB; Calogero *et al.*, 1980) are shown in yellow. (*b*) Clicking on a hydrogen bond shows the full molecule at the other end, enabling a sheet of hydrogen-bonded molecules (not shown) to be built up quickly.

In addition to this core functionality, *Mercury* provides a range of file input and export options, including the ability to load and browse through all or a subset of the entries in the Cambridge Structural Database (CSD). Distances, angles and torsions can be measured. Most options are available both through menu buttons permanently available in the main program interface and *via* a context-sensitive right-mouse-button pull-down menu.

3. Program enhancements

3.1. Multiple structure display and overlay

In order to aid comparison of crystal structures, it is now possible for two or more structures to be displayed simultaneously in the visualization area. A control box lists all structures on display and allows the colour of each individual structure to be changed so that they may easily be distinguished (Fig. 3). Any given structure may be linked to or decoupled from the movement of the mouse, which means that structures may be moved relative to one another and, in particular, manually overlaid. Critical to this operation is the ability to switch at will between local and global rotation centres, so that structures are rotated around their individual centroids, or around the centroid of all structures, depending on the user's requirement. (The positions of rotation centres can also be customized, *e.g.* placed on a selected atom.) Since manual overlay can be tedious, it is also possible to overlay any two structures by least-squares fitting of selected pairs of atoms. This type of functionality is helpful when comparing the structures of related molecules, or different polymorphic forms of the same chemical compound. For example, Fig. 4 illustrates an overlay of the monoclinic and orthorhombic forms of naphthalene tetroxide imine, and shows that the two forms contain similar, but not quite identical, hydrogen-bond motifs.

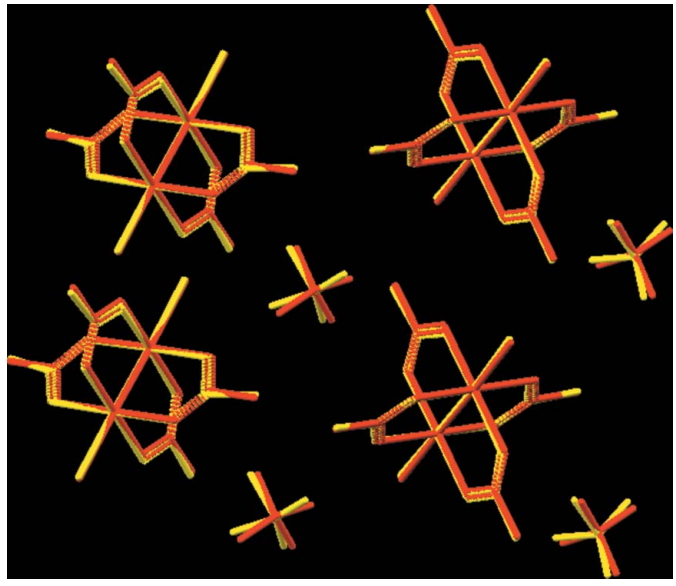


Figure 3
Overlay of a space-group reinterpretation in *C*2/*c* [CSD refcode AACRUB01 (Marsh & Schomaker, 1981); shown in red] and the original crystal structure in *Cc* [CSD refcode AACRUB (Bino *et al.*, 1979); shown in yellow].

3.2. Displacement ellipsoids

Atomic displacement parameters (ADPs) can now be shown as displacement ellipsoids, *via* the *Ellipsoid* display style (Fig. 5), the implementation being based on a published methodology (Grosse-Kunstleve & Adams, 2002). ADPs can be read from CIF and *SHELX* res files. Control is possible over the probability level and (for anisotropically refined atoms) whether principal ellipses are to be drawn. H atoms with ADPs can be drawn as fixed-size spheres. Atoms with non-positive-definite ADPs can be displayed as cubes, enabling their easy identification.

3.3. General graphical display improvements

Depth cueing is now available to aid three-dimensional perception; this is particularly important when viewing large packing arrays comprising several unit cells. Several other display enhancements are more to do with the preparation of suitable two-dimensional images for publication than for facilitating interactive use of the program. Control is possible over ambient and directional lighting effects and multi-colour gradient backgrounds can be set up; this allows the creation of highly customizable images (*e.g.* Fig. 6). The size and colour of atom labels can be customized, and an algorithm has been devised for positioning labels so as to minimize their overlap with

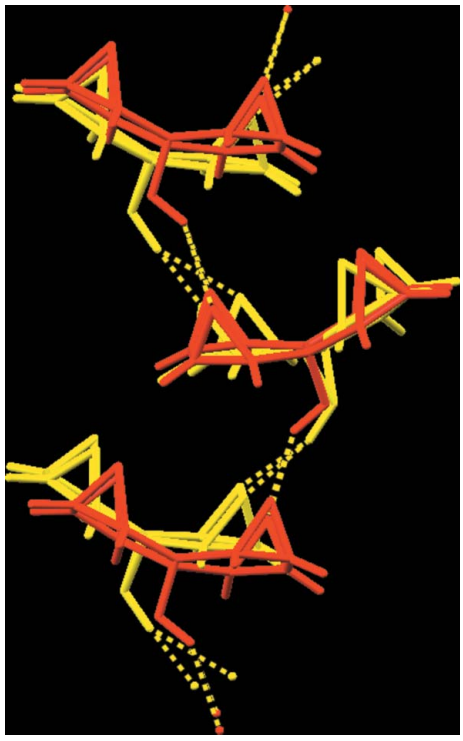


Figure 4
Similar but different hydrogen-bond motifs in the orthorhombic (CSD refcode BAZYAC; shown in yellow) and monoclinic (CSD refcode BAZYAC01; shown in red) polymorphs of naphthalene tetroxide imine. Both structures were reported by Liu & Davis (1981).

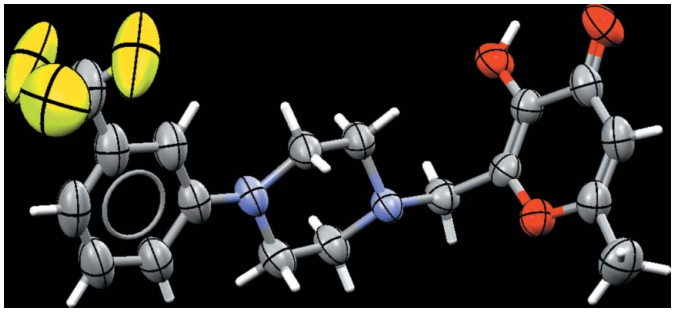


Figure 5
Display of atomic displacement ellipsoids in *Mercury*. [The molecule was reported by Köysal *et al.* (2004).] The non-spherical ADPs of the disordered CF_3 group are clearly visible.

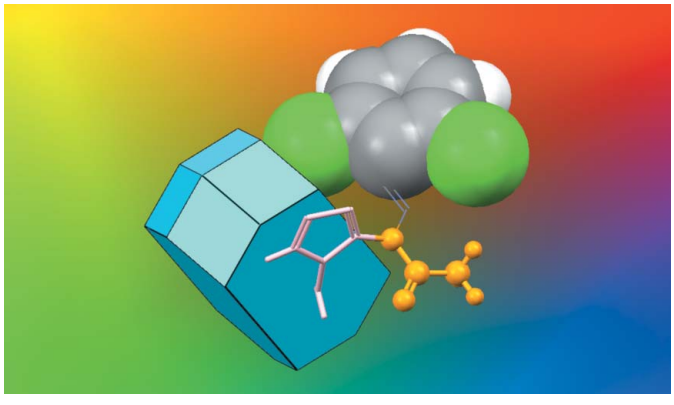


Figure 6
Example of how most aspects of the *Mercury* display can be customized.

atoms. Display of bond types and aromatic rings can be turned on or off. The colours of individual contacts and unit-cell axes can be controlled.

3.4. Measuring capabilities

These are little changed since the original release. A few minor improvements have been made; for example, it is now possible to measure geometrical parameters involving planes and centroids.

3.5. Structure input and editing

Any available databases in the Cambridge Structural Database format (ASER) are automatically loaded at start time. Additionally, *Mercury* can load lists of structures found by searching the CSD or in-house ASER databases with the search program *ConQuest*, either by launching *Mercury* from within *ConQuest* or by means of an intermediate list of refcodes stored in a gcd file. *Mercury* reads data from a variety of file formats (CIF, mol2, pdb, *SHELX* res, sdf) and writes CIF, mol2, pdb and res files. Since several of the file formats do not carry bond-type information, an algorithm has been implemented for deducing bond types from three-dimensional atomic coordinate information (Bruno *et al.*, 2004). This algorithm is completely successful for about 85% of structures. Manual bond-type editing options are available for correcting the occasional mistakes made by the automated algorithm. It is also possible to add missing H atoms, add new bonds, and remove atoms and molecules.

3.6. Display of auxiliary information

If the information is available (*i.e.* has been read in from the input file or CSD entry), *Mercury* can list a wide variety of textual information relating to displayed structures, for example, *R* factor, cell and reduced cell dimensions, compound name and literature reference. In the case of structures read from the CSD, a chemical diagram will normally be available for display.

3.7. Spreadsheet views

Data relating to the three-dimensional display can be tabulated in spreadsheet form. For example, the *Atoms* spreadsheet can be used to list a customizable selection of parameters for every atom on display, including label, colour, fractional coordinates, ADPs and the symmetry operation used to generate the atom from the original asymmetric unit (Fig. 7). Similar spreadsheets can be produced for bonds, non-bonded contacts, centroids, planes, symmetry operators, and any distances, angles and torsions that may have been measured. The contents of spreadsheets may be exported in comma-, tab- or space-separated formats. A spreadsheet can be sorted on the values in any chosen column; for example, this method could be used to identify rapidly the shortest contacts on the display. Clicking with the right-hand mouse button on a row produces the same menu as would have been obtained by right-clicking on the corresponding object in the three-dimensional display area. Any row of the *Atoms* spreadsheet can be selected (or multiple rows, *e.g.* by clicking and dragging). This functionality opens up some useful ways of interacting with the three-dimensional display; for example, all atoms generated by a particular symmetry operation could be given the same colour by sorting on the values in the symmetry-operation column, selecting the relevant rows (which, after the sort, will be contiguous), and right-clicking to access the *Colour* menu.

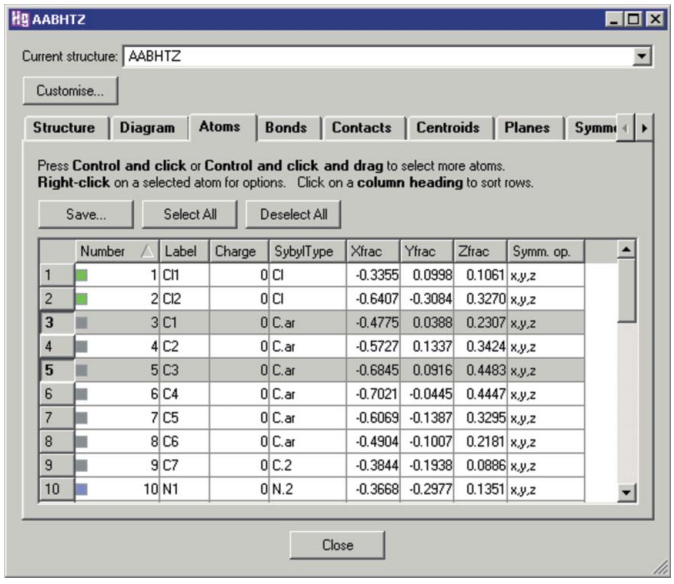


Figure 7
Spreadsheet view of CSD refcode AABHTZ. Selecting rows 3 and 5 as shown automatically selects the corresponding atoms in the three-dimensional visualizer, and *vice versa*.

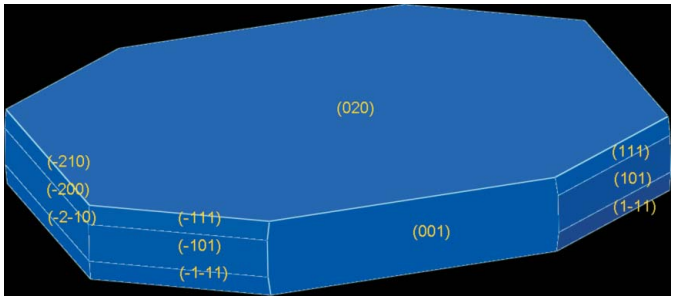


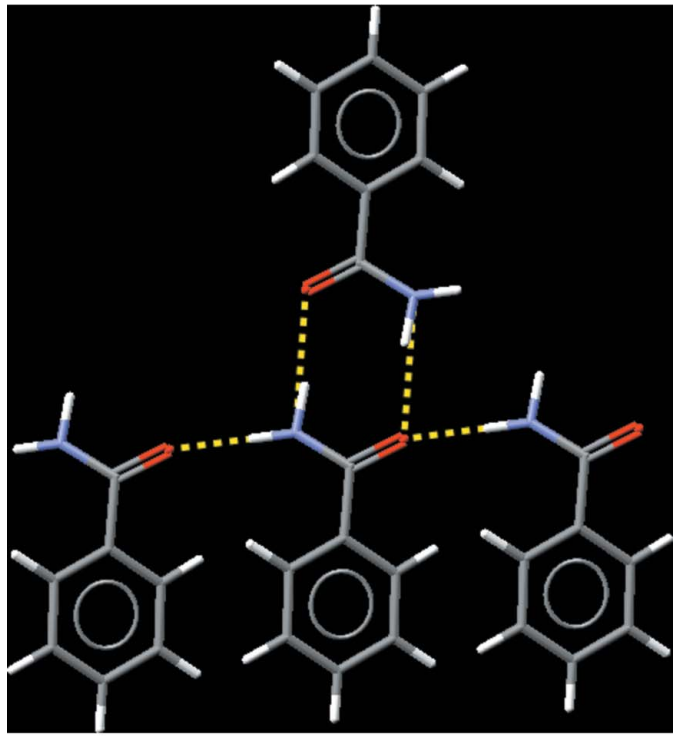
Figure 8
Bravais–Friedel–Donnay–Harker morphology of *n*-heneicosane (CSD refcode ZZZOQO; Smith, 1953).

3.8. Simulated powder patterns

Powder-diffraction patterns can be simulated and displayed, with tick marks to indicate the expected positions of reflections and systematic absences. Any part of the pattern can be magnified. Placing the cursor at any position on the plot will cause the *hkl* indices of the nearest reflection to be displayed. The profile can be saved to a file in xye format, containing three columns for 2θ, intensity and the estimated standard deviation of the intensity, calculated as the square root of the intensity. In the simulation, the Lorentz–polarization correction assumes a laboratory X-ray source and fixed slit widths. No absorption is simulated and no noise or background included. All non-H atoms are assumed to have isotropic ADPs (U_{iso}) of 0.05 Å^2. H atoms for which three-dimensional coordinates are available may optionally be taken into account, in which case they are assigned U_{iso} values of 0.06 Å^2. Correction for partial occupancies due to atoms at special positions is taken care of automatically; correction for partial occupancies due to disorder is handled correctly if this information is available. All reflections have a symmetric pseudo-Voigt peak shape with an FWHM of 0.1° 2θ, corresponding to medium-resolution laboratory data. The 000 reflection is excluded. The default 2θ resolution is 50.0°, which, for the default Cu $K\alpha_1$ radiation, corresponds to a direct-space resolution of 1.8 Å.

```
#include "AserDatabase.h"
#include "CrystalMaker.h"
#include "HBondCriterion.h"
#include "Mol2File.h"

#include <iostream>

int main()
{

    // Open a database
    HCrystalStructureDatabase database = new AserDatabase("C:\Program
Files\CCDC\CSD V5.27\as527be");

    // Extract Benzamide
    HCrystalStructureDatabaseEntry entry = database->entry(
DatabaseEntryIdentifier("BZAMID") );

    // Write the crystal structure out
    HCrystalMaker crystal_maker = entry->crystal_maker();
    std::cout << *crystal_maker << std::endl;

    // Create a definition of H-Bonds
    HContactCriterion hbond = new HBondCriterion(0.0);

    // Find all short contacts in our crystal
    HCrystal crystal = crystal_maker->create_nonbonded_molecule(*hbond);

    // Save the arrangement of molecules as a Mol2 file
    Mol2File file;
    file.add(*crystal, DatabaseEntryIdentifier("BZAMID"));
    file.print("BZAMID_contacts.mol2");

    return 0;
}
```

(*a*)



(*b*)

Figure 9
(*a*) The main body of a C++ program for finding hydrogen bonds in the crystal structure of benzamide (CSD refcode BZAMID). The main body consists of calls to the underlying C++ libraries that do the actual work. (*b*) The output of the C++ program in (*a*) viewed in *Mercury*.

3.9. BFDH morphologies

It is now possible to predict and display crystal morphologies using the Bravais–Friedel–Donnay–Harker (BFDH) method (Fig. 8), which was implemented following published methodology (Bennema & Meekes, 2004). This method is crude, the prediction being based only on the cell dimensions and crystal symmetry. However, the code for generating the morphology is clearly separated from that for

predicting the relative growth rate of possible crystal faces, so more sophisticated algorithms for the latter should be easy to plug in. *Mercury* observes the space-group selection rules and therefore correctly displays non-relative prime Miller indices [*i.e.* (020) instead of (010)] where applicable.

4. Program design

Mercury is written in object-oriented (OO) C++. Memory is allocated on the fly, and the number of atoms, number of reflections in a powder diffraction pattern, number of crystal structures, *etc*. are limited only by the specifications of the hardware. The graphics display and graphical user interface rely on third-party libraries (OpenGL, Qt) and are clearly separated from the underlying functionality. The core scientific and mathematical functionality is provided by several libraries developed collectively by CCDC programmers since 1998. These libraries cover areas such as CIF handling, chemical-diagram manipulation, basic crystallography and chemistry, chemical analysis, substructure searching, space-group and point-group symmetry, powder diffraction, general mathematics, databases, and text formatting and display. Throughout these libraries, the matrix representation of space-group symmetry operators is used where available to avoid space-group setting ambiguities associated with space-group names, *cf.* choice of origin ambiguities for space group *Fddd*. Fig. 9(*a*) shows a segment of code for finding hydrogen bonds in the crystal structure of benzamide (Penfold & White, 1959), Fig. 9(*b*) illustrating the sort of display that might thus be produced in *Mercury*. Finally, our VisLib library converts basic crystallographic objects to objects suitable for graphical display; VisLib is used in several other CCDC products, such as *Mogul*, *enCIFer*, *Relibase+* and *SILVER*. Our extensive use of Qt (Trolltech AS, 2005) adds greatly to the ease with which functionality can be exploited in several different applications and has helped us begin to establish a common look-and-feel across our products.

Mercury remains under continuous development and recent work has focused on improving the OO design of VisLib so that it can be used more flexibly. For example, colouring 'policy classes' have been introduced, and these should facilitate highly customizable ways of colouring and labelling atoms and bonds. Classes for generating and manipulating grids, contouring algorithms, and surfaces have been written. These, combined with recent changes in our use of OpenGL's display list facility, are enabling steps towards the display of both opaque and semi-transparent surface objects (which, in fact, are already available in another VisLib-based application for displaying protein cavities in the CCDC program *Relibase+*). In summary, both the design and the functionality of *Mercury* are still under active development, and we expect the program to be further enhanced over the next few years.

5. Documentation, availability and environment

Mercury is fully documented, and there are several tutorials. Two versions of the program are available: one can be freely downloaded from the CCDC website (http://www.ccdc.cam.ac.uk/); the other is provided to users of the Cambridge Structural Database system. Some items of functionality (display and overlay of multiple structures, editing of molecules, and BFDH morphology prediction) are restricted to the latter version. Platforms on which *Mercury* is supported include Windows (Intel compatible, Windows 2000/XP), Linux (Intel compatible, 32bit: Red Hat Linux 7.3, 8, 9; Red Hat Enterprise 3.0, 4.0; SuSe 8.1, 8.2, 9; Mandrake 9.2, 10; Mandriva 10.2; Debian 3.0), Silicon Graphics R5000 and above (minimum of IRIX 6.5.13), Sun SPARC (Solaris 8, 10), and Mac OSX 10.3.

Thanks are due to many CCDC programmers whose code has been exploited by *Mercury*, especially Jason Cole, Lucia Rodriguez-Monge, Lucy Purkis and Ian Bruno, and to the members of CCDC's technical and scientific support groups, especially Karen Cooke, who implemented the Mac OSX port, and Gary Battle, who has been responsible for much of the documentation.

References

Bennema, P. & Meekes, H. (2004). *Nanoscale Structure and Assembly at Solid-Fluid Interfaces*, edited by X. Y. Liu & J. De Yoreo. Boston: Kluwer Academic Publishers.
Bino, A., Cotton, F. A. & Felthouse, T. R. (1979). *Inorg. Chem.* **18**, 2599–2604.
Bruno, I. J., Cole, J. C., Edgington, P. R., Kessler, M., Macrae, C. F., McCabe, P., Pearson, J. & Taylor, R. (2002). *Acta Cryst.* B**58**, 389–397.
Bruno, I. J., Cole, J. C., Kessler, M., Luo, J., Motherwell, W. D. S., Purkis, L. H., Smith, B. R., Taylor, R., Cooper, R. I., Harris, S. E. & Orpen, A. G. (2004). *J. Chem. Inf. Comput. Sci.* **44**, 2133–2144.
Calogero, S., Russo, U. & Del Pra, A. (1980). *J. Chem. Soc. Dalton Trans.* pp. 646–653.
Grosse-Kunstleve, R. W. & Adams, P. D. (2002). *J. Appl. Cryst.* **35**, 477–480.
Köysal, Y., Isik, S. & Aytemir, M. D. (2004). *Acta Cryst.* E**60**, o112–o114.
Liu, L.-K. & Davis, R. E. (1981). *Bull. Inst. Chem. Acad. Sin.* **28**, 49–57.
Marsh, R. E. & Schomaker, V. (1981). *Inorg. Chem.* **20**, 299–303.
Motherwell, W. D. S., Shields, G. P. & Allen, F. H. (1999). *Acta Cryst.* B**55**, 1044–1056.
Penfold, B. R. & White, J. C. B. (1959). *Acta Cryst.* **12**, 130–135.
Smith, A. E. (1953). *J. Chem. Phys.* **21**, 2229–2231.
Taylor, R. & Macrae, C. F. (2001). *Acta Cryst.* B**57**, 815–827.
Trolltech AS (2005). Qt. Version 3.3.3. Trolltech AS, Oslo, Norway. http://www.trolltech.com/.