5.3.2.1.3. *Limitations of vcif*

Because the program is testing certain properties of character strings within logical lines of a file, it stores a line at a time for further internal processing. If a line contains a null character (an ASCII character with integer value zero), this will be taken as the termination of the string currently being processed, according to the normal conventions in the C programming language for marking the end of a text string. In this case, subsequent error messages may not reflect the real problem. The null character, of course, is not allowed in a CIF.

vcif also interprets syntax rules literally, so a misplaced semicolon might mean that a large section of the file is regarded as a text field and too many or too few error messages are generated. This can make a correct interpretation of the causative errors difficult for a novice user.

5.3.3. Editors with graphical user interfaces

A useful class of editing tool is the graphical editor, where different types of access can be provided through icons, windows or frames, menus and other graphical representations. The availability of standard instructions through drop-down menus makes such tools particularly suitable for users who are not expert on the fine details of the file format. The ability within the program to restrict access to particular regions of the file makes it easier to modify the contents of a CIF without breaking the syntax rules. A small but growing number of such editors are becoming available, such as those described here.

5.3.3.1. *enCIFer*

The program *enCIFer* (Allen *et al.*, 2004) has been developed as a graphical utility designed to indicate clearly to a novice user where errors are present in a CIF, to permit interactive editing and revalidation of the file, and to allow visualization of three-dimensional structures described in the file. In its early releases, it was targeted at the community of small-molecule crystallographers interested in publishing structures or depositing them directly in a structure database. Version 1.0 depended on a compiled version of the CIF core dictionary, but subsequent versions allow external CIF dictionaries to be imported. At the time of publication (2005), development is concentrating on support for DDL1 dictionaries.

Given its target user base, the purpose of the program is to permit the following operations within single- or multi-block CIFs:

(i) Location and reporting of syntax and/or format violations using the current CIF dictionary.

(ii) Correction of these syntax and/or format violations.

(iii) Editing of existing individual data items or looped data items.

(iv) Addition of new individual data items or looped data items.

(v) Addition of some standard additional information *via* two data-entry utilities prompting the user for required input ('wizards'): the *publication wizard*, for entering the basic bibliographic information required by most journals and databases that accept CIFs for publication or deposition; and the *chemical and crystal data wizard*, for entering chemical and physical property information in a CIF for publication in a journal or deposition in a database.

(vi) Visualization of the structure(s) in the CIF.

In all cases where data are edited or added, *enCIFer* can be used to check the format integrity of the amended file.

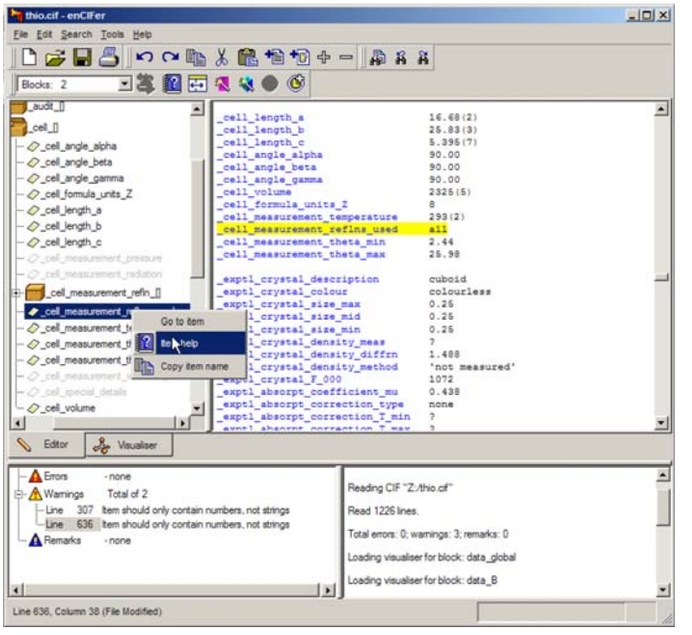

Fig. 5.3.3.1. The *enCIFer* graphical user interface.

5.3.3.1.1. *The main graphical window*

Fig. 5.3.3.1 is an example of the use of *enCIFer* to read and modify a CIF. The figure shows the components of the main window after a file has been opened. Beneath the standard toolbar that provides access to operating-system utilities and to the main functions of the program itself is a task bar (here split over two lines) providing rapid access to a subset of the program's features. Under this are two large panes. The pane on the right is the editing window, where the content of the CIF is displayed and may be modified. The left-hand pane is a user-selectable view by category of the data names stored in the CIF dictionary against which the file is to be validated. At the bottom are two smaller panes. The one on the right logs the session activities and displays informational messages. The left-hand pane lists errors and warning notices generated by the validation system. Errors are labelled by line number, and selection of a specific message (by a mouse double-click) scrolls the content of the main text-editing window to that line number.

Tabs in the middle of the display allow the user to switch rapidly between the editing mode and a visualization of the three-dimensional structures described in the CIF.

These components are described more fully below, followed by a description of the other windows that may be created by a user: the help viewer, the loop editor and the data-entry wizards.

5.3.3.1.2. *The interface toolbar*

This toolbar provides menus labelled 'File', 'Edit', 'Search', 'Tools' and 'Help' that provide the expected functionality of graphical interfaces: the ability to open, close and save files, store a list of recently accessed files, spawn help and other windows, allow searching for strings within the document, allow the user to modify aspects of the behaviour or the look and feel of the program, and provide entry points for specific modes of operation. The most useful of these utilities can also be accessed from icons on the task bar. They are discussed in more detail in the following section.

This main menu is structured in a way familar to users of popular applications designed for the Microsoft Windows operating

system, although the *enCIFer* program runs on a variety of different operating systems and machine hardware platforms. Nevertheless, the use of a common menu style makes the initial use of the program much easier for novice users and allows the program to be effectively used without detailed study of its documentation.

5.3.3.1.3. *The task bar*

The task bar allows rapid one-click access to the standard operations of creating a new document, opening, saving or printing the contents of the current file, copying, cutting and pasting text, searching for specific text within the document, and undoing or redoing previous edits.

Two buttons allow insertion of complete text files. One allows the user to select any file from local or network-mounted file systems. The other imports a specific file (the location of which may be specified by the user through the 'Preferences...' selection of the main 'Edit' menu). While this specific file may contain anything, it is intended to be a template CIF that a user will tailor to meet their own requirements. The default provided with the software is a standard template distributed by the IUCr for use in submitting articles to *Acta Crystallographica*. In either case, the file is imported at the current editing location and is not subject to validation upon input; the user must manually revalidate the file after import.

An icon on the task bar allows the user to run a validation procedure. This icon will be dimmed (indicating that the validation procedure may *not* be run) unless the user has modified the contents of the CIF. Other icons on the task bar behave in the same manner, allowing the procedures with which they are associated to be executed only under appropriate circumstances. Thus, for example, the looped list editor is not invoked unless the user clicks within the reserved word `loop_` in a list header.

Similarly, the 'help' icon in the task bar is dimmed unless the user has selected a data name in the CIF; when this is done, the icon is activated and clicking on it launches a help window containing the CIF dictionary definition of the data item.

The task bar also contains a drop-down menu listing all the data-block names in the current file. When the user selects one of the data-block names, the edit cursor is positioned at the head of the matching data block in the edit window. This is a rapid and efficient way of navigating within large and complex files.

The other buttons provided on the task bar allow the user to: reduce or increase the font size in the editing window; create a new looped list within the loop-editing window; invoke the publication and data-entry wizards; and hide or reveal the dictionary browse window pane.

Users may modify the appearance of the task bar to retain or conceal subsets of these icons, depending on which they find most useful.

5.3.3.1.4. *The main edit pane*

The main edit pane is a text-editing area where the user may directly modify the content of a CIF. Colours and font styles are used to indicate different syntactic elements. The details of the colours and styles may be modified to suit the user.

For the novice user, this is perhaps the most immediately helpful feature offered by this program. When a trailing semicolon is inadvertently lost from an extended text field, typical sequential parsers may interpret succeeding tokens as part of the quoted text and produce misleading error reports. Within the *enCIFer* edit window, all such text is marked up in a specific colour (green by default) so that the fault is much more obvious to the human eye and its source much easier to locate.

Two other typographic cues are used to help the user to trace errors, or to ensure that certain text has been input correctly. Subscripts and superscripts are represented in a smaller typeface (and in a different colour) so that missing delimiter characters are again obvious to the eye. Secondly, some special characters in the conventional CIF encoding (such as Greek letters) are displayed in an appropriate symbol font when the file is first loaded, so that for example the input string `\a` is rendered as α. Note that the backslash character is retained, and that the symbol character is not generated as new text is input or edited. This scheme therefore has some potential for confusion, but is nevertheless helpful in checking that less obvious special codes have been entered correctly.

The user is free to enter arbitrary text in this pane, possibly breaking CIF syntax rules in the process. Only when the revalidation process is manually invoked will the file be rescanned and any errors reported.

5.3.3.1.5. *The dictionary browse pane*

The upper left-hand pane in Fig. 5.3.3.1 illustrates the dictionary browser, an optional graphical view of the contents of the CIF dictionary against which the file is being validated. (The presence or absence of this pane is toggled from an icon in the task bar.) Box icons represent the contents of categories, and the tree of category containers may be expanded or collapsed as desired to show individual items within categories.

A dictionary view is generated for each separate data block in the CIF. Within the dictionary view of an individual data block, those data items present in the data block are shown in bold; other items defined in the dictionary but absent from the current data block appear in a lighter colour.

Within the dictionary browse pane, a user may select (with a click of the appropriate mouse button) a menu of three options which depend on whether the data name is present or absent in the data block. If present, one option positions the cursor in the editing window at the location of the selected data item. If the item is absent from the data block, the user is given the option to paste the data name into the editing window at the current insertion point. The other options (in both cases) are to copy the data name to the clipboard or to open the help window with the CIF dictionary definition of the selected item.

5.3.3.1.6. *The error notification pane and logging area*

The lower left-hand pane of Fig. 5.3.3.1 illustrates typical error notices generated by the parser when the validation process is invoked. At present, the classification of the severity of errors is guided by the editorial requirements of databases and journals, and does not necessarily match the formal errors dictated by the CIF specification. It is likely that this will change in future releases as validation is driven increasingly by the dictionaries rather than by hard-coded subroutines.

A convenient feature is that double-clicking on the line number in the error report relocates the cursor to that line in the editing pane. At present, error messages are listed by line only – they are not grouped by data block.

The user has a small number of options to control error notification. The choice of the maximum number of consecutive error lines to permit before error checking is abandoned is a useful way, especially for novices, to reduce the amount of output generated by severe syntax errors and to focus on repairing individual errors. The user may also specify a file that contains a set of CIF data names which are considered *mandatory* components of a particular file. Absence of any of these items from the current data

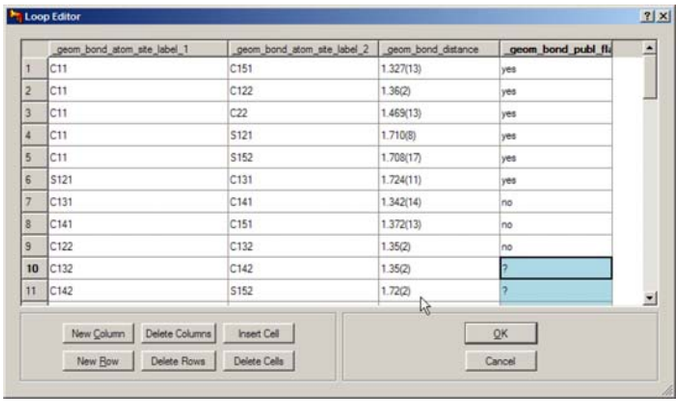

Fig. 5.3.3.2. The *enCIFer* loop editor.

block is flagged as an error. The program log in the lower right-hand part of the program window records the history of the user's interactions with the file during the current editing session.

Information is written to the status bar (the lower margin of the window) to indicate the location by line and column number of the editing cursor.

5.3.3.1.7. *The loop editor*

The program has a useful spreadsheet-style editor for looped lists (Fig. 5.3.3.2). A particular benefit of this style of display is that the spreadsheet cells are arranged in a rectangular grid, so that visual scans can often detect deviations from a pattern of values within a column, thus making it easy to identify placement errors where values have been omitted or inadvertently conjoined. Such errors are not always obvious by direct visual inspection of a CIF, where the layout of a looped list need not follow any regular pattern.

The buttons to add or delete columns allow for the straightforward addition or deletion of data items from the loop. If the user selects the 'New Column' button, a small pop-up window helpfully provides a view of the associated dictionary (in the same hierarchical category-based tree view of the dictionary browser pane) to help the user select the required new data name. The 'Insert Cell' and 'Delete Cells' buttons are convenient tools for the realignment of rows and columns where values have been omitted or misplaced.

The loop editor is invoked from one of two buttons in the task bar, allowing either the creation of a new looped list or the modification of an existing one. As with the application as a whole, there is no dynamic validation of input; the new list must be saved and the entire CIF then manually revalidated.

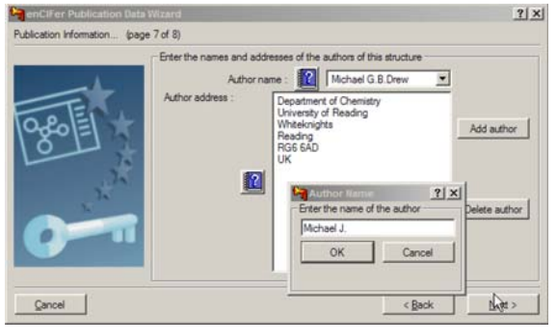

Fig. 5.3.3.3. The *enCIFer* publication data wizard. Information about the title and authors of an article to be submitted for publication is requested through a sequence of linked dialogue boxes.

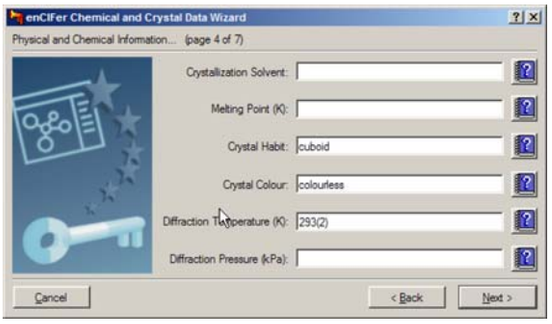

Fig. 5.3.3.4. The *enCIFer* chemical and crystal data wizard.

5.3.3.1.8. *The publication and chemical and crystal data wizards*

The user may invoke data-entry 'wizards', subordinate programs that prompt for particular data items useful for the publication of a crystal structure report or for the deposition of a crystal structure in a database. This is the kind of information that might be requested in the *Notes for authors* for a journal, and it is helpful if the information is routinely requested from inexperienced authors during normal use of the software. The data-entry tools are known as 'wizards' because they will utilize information already in the file.

Hence, as shown in Fig. 5.3.3.3, details of an article's contact author are retrieved from the CIF and used to seed a list of contributing authors. As the address for each author is entered, the program makes each new address available as a stored record for easier input of additional information.

Fig. 5.3.3.4 demonstrates the same approach to encouraging authors to supplement information already in the CIF with related chemical (or crystal) data not usually provided by the CIF generators embedded in crystallographic structure determination programs.

5.3.3.1.9. *The visualization window*

A final useful feature of *enCIFer* is its ability to visualize the three-dimensional structure of molecules described in the data blocks of a CIF. Fig. 5.3.3.5 demonstrates crystal packing with

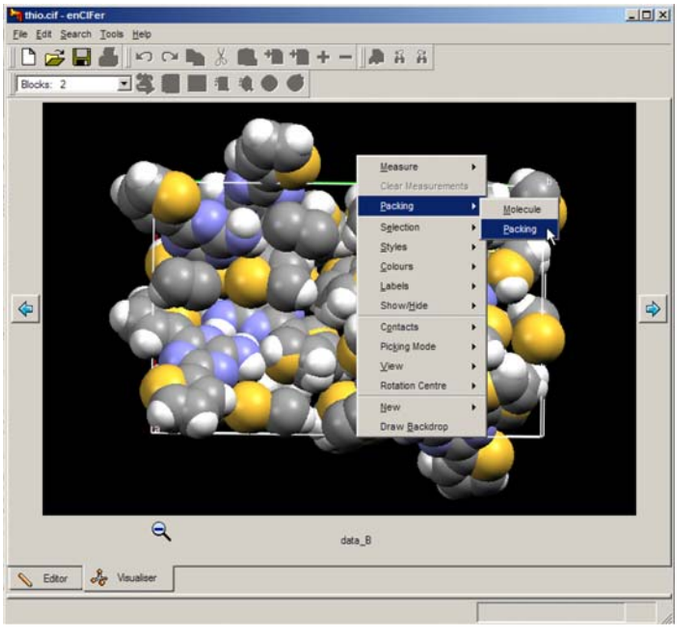

Fig. 5.3.3.5. Visualization of a molecular and crystal three-dimensional structure with *enCIFer*.

a space-filling molecular representation, and the drop-down menu indicates some of the options available to modify the appearance of the graphics. The molecular-graphics library used by the program is part of the larger database interface software package developed at the Cambridge Crystallographic Data Centre. In the present version, the visualizer is run only upon initial parsing of the input CIF, and therefore does not provide an ability to track visually the molecular changes associated with direct modification of the contents of the file.

5.3.3.2. *CIFEDIT*

The *CIFEDIT* program (Toby, 2003) is written in Tcl/Tk (Ousterhout, 1994) and provides an application for viewing and editing CIFs. The code is written in such a way that it can be embedded into larger programs to provide a CIF-editing interface within larger application suites.

The current version of the program is able to validate CIFs against both DDL1 and DDL2 dictionaries, although the DDL2 validation is currently less complete than for DDL1. For example, numeric values are checked against permitted enumeration ranges only for DDL1. Dictionaries are accessed through index files, each of which contains Tcl data structures that point to the location of the definitions in the dictionary file itself and store information such as units and enumeration ranges that can be used for data validation. A utility provided with the program allows a user to generate new index files when new versions of the dictionaries become available. It is intended that dictionary indexing will be incorporated within the main application in the next program release, so that interactive dictionary selection will be possible.

When a CIF is opened, the contents are parsed and validated against one or more user-selected dictionaries. Errors are displayed in a pop-up window and may be written to a file or viewed within the application. The main program window displays the contents of the CIF in two primary panes (Fig. 5.3.3.6). In the left-hand pane, a tree structure shows the data blocks in the file and the data names present in each block. The data blocks may be expanded or collapsed by the user, to present an overview or a detailed view of the data structure of the file. Underneath the icon representing the data block, non-looped data items are listed alphabetically. The figure demonstrates how a single value may be selected in the left-hand pane (`_cell_length_a`) and displayed in the main window. Physical units for the selected quantity are extracted from the corresponding dictionary definition and presented alongside the numeric value. The dictionary definition may also be displayed in a separate pop-up window using the 'Show CIF Definitions' button.

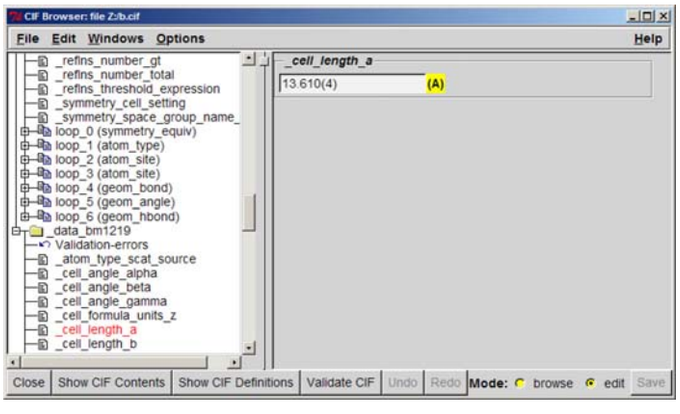

Fig. 5.3.3.6. The use of *CIFEDIT* to display and alter the contents of a CIF; here a non-looped data item is shown.

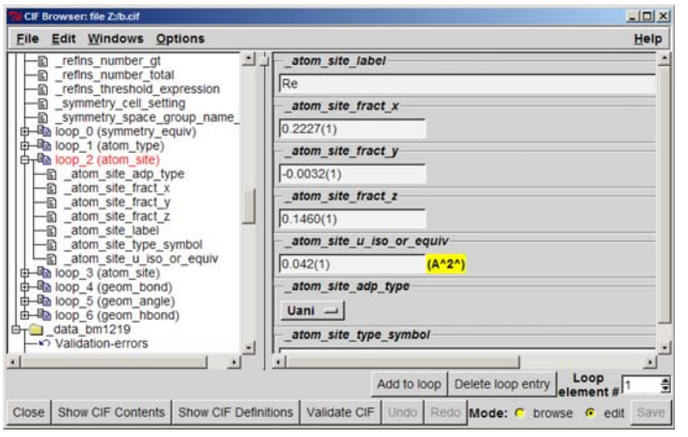

Fig. 5.3.3.7. Row-based loop editing with *CIFEDIT*; here loop_2 (comprising the ATOM_SITE category) has been selected by the user; the editing cursor begins at row 1 of the loop.

The program may be run in two modes: a 'browse' mode, where the selected value is displayed in the main pane, but may not be altered; and an 'edit' mode (as in the example) where the value appears in an editable text widget.

Data loops in the CIF are displayed after the alphabetical list of non-looped items. The loops are numbered sequentially from zero and an indication of the loop category is given in parentheses in the tree-view window. The loop 'branches' of the tree may be expanded or collapsed as the user wishes.

Loops may be viewed and edited in two ways: by row or by column. If the user selects the loop title node in the hierarchical view pane, the loop is presented by row, starting in sequence at row 1 (Fig. 5.3.3.7). Other rows may be selected by using the address box in the lower-right-hand part of the window. Alternatively, if the user selects an individual data name within the loop representation in the hierarchical view, all instances of that data item within the loop are displayed in the main pane. (In practice the number of values shown is constrained to a maximum number that the user may choose, so that the application does not run out of memory if there are very large loops.)

For items with a restricted set of permitted values in the dictionary, the editing function allows the user to select only one of the permitted options *via* a drop-down menu.

While the application is intended to be used in this structured and itemized mode, there is an option to open the entire CIF in a text-editing window if there are errors that cannot be handled in the normal mode. This is not recommended, but is occasionally convenient. While this free-text editing mode is in operation, the ability to modify the file through the structured editing pane is suspended to avoid conflicting changes.

After any change has been made, the user may revalidate the file. This is strongly recommended after making changes in the free-text editing mode.

5.3.3.3. *HICCuP*

The program *HICCuP* (Edgington, 1997) was an early graphical utility developed at the Cambridge Crystallographic Data Centre for interactive editing and validation of a CIF. It is no longer supported, having been replaced by *enCIFer* (Section 5.3.3.1). Nevertheless, it contained some interesting features and is of potential interest to developers using multiple-platform scripting languages. It was implemented in the Python language (van Rossum, 1991) and required that Tcl/Tk (Ousterhout, 1994) be also available on the host computer. The name of the program is an acronym for 'High-Integrity CIF Checking using Python'.