

## 5.3. SYNTACTIC UTILITIES FOR CIF

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.

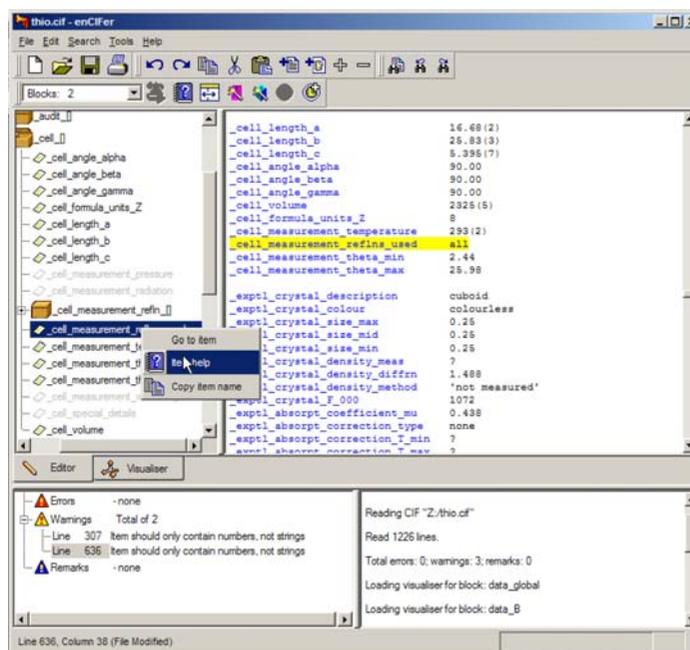


Fig. 5.3.3.1. The *enCIFer* graphical user interface.

## 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 familiar 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  $\alpha$ . 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