Journal of
Applied Crystallography
ISSN 0021-8898

Received 2 April 2008
Accepted 8 May 2008

VCIF2: extended CIF validation software

Georgi Todorov[a]* and Herbert J. Bernstein[b]

[a]95 Biltmore Avenue, Oakdale, NY 11769, USA, and [b]Department of Mathematics and Computer Science, Dowling College, 150 Idle Hour Boulevard, Oakdale, NY 11769-1999, USA. Correspondence e-mail: terahz@geodar.com

Recent revisions to the CIF standard, the growing number of dictionaries and the critical role played by CIF in the IUCr publication process led the IUCr to fund a two-year project to upgrade portions of the existing CIF software base to support longer lines and more rigorous validation of CIFs against multiple layered dictionaries. A database-based approach to validation to ensure compliance with data-range and enumeration specifications, to ensure compliance with parent–child relationships, and to detect missing and duplicated tags is presented here. This approach to validation is being extended to support the handling of binary synchrotron imgCIF data.

1. Introduction

'The term 'crystallographic information file' (CIF) refers to data and dictionary files conforming to the conventions adopted by the IUCr in 1990 and revised by the IUCr Committee for the Maintenance of the CIF Standard (COMCIFS). The CIF format is intended to meet the needs of a wide range of scientific applications within, and without, the discipline of crystallography.' (Hall *et al.*, 2005.) Validation of a CIF involves two checks: one check for the syntax (*i.e.* checking against the formal rules of the grammar) and then a check for the domain content. In this paper we are concerned with checking the syntax. Such syntax validation is carried out by many existing programs, for example the program *VCIF* (McMahon, 1998, 2005). CIF has grown over the years and a revised version of *VCIF*, *VCIF2*, has become necessary.

We present a software suite for easy extended lexical, parser and dictionary CIF validation for IUCr publications. The chemical and biological content are not validated except in the presence of and to the extent provided by a dictionary. *Acta Crystallographica Section C: Crystal Structure Communications* (http://journals.iucr.org/c/) and the more recent *Section E: Structure Reports Online* (http://journals.iucr.org/e/) are two popular journals of the IUCr used for publishing crystal structures. *Section F: Structural Biology and Crystallization Communications Online* (http://journals.iucr.org/f/) is a new journal of the IUCr. All three journals use the crystallographic information framework (CIF; Hall *et al.*, 1991) for submission of new structures and require careful validation of the CIFs. Recently, so-called long-line CIFs have been introduced for *Acta C* and *E* and new software for handling long lines was required. For *Acta F* there is a need for validation programs that can handle the complexity of mmCIF (see Appendix *A* for a glossary of terms). mmCIF introduces parent–child relationships among categories (tables) that require extensions to the existing validation software. The parent–child relationship (a child is a subtable and the parent is the corresponding supertable) is one of the key features of a relational database. Hence, we introduce a new generation of validation software that uses the database model for dictionary validation without needing a local database server. The software is available as a server web page and as a downloadable kit.

```
data_1ZRT
#
loop_
_audit_author.name
'Berry, E.A.'
'Huang, L.S.'
'Saechao, L.K.'
'Pon, N.G.'
'Valkova-Valchanov, M.'
'Daldal, F.'
#
_pdbx_database_status.status_code                REL
_pdbx_database_status.entry_id                   1ZRT
_pdbx_database_status.recvd_deposit_form         N
_pdbx_database_status.date_deposition_form       ?
_pdbx_database_status.recvd_coordinates          Y
_pdbx_database_status.date_coordinates           2005-05-21
```

Figure 1
Beginning of `1zrt.cif`.

2. CIF, CBF and imgCIF

The acronym CIF is used both for the crystallographic information file, the data exchange standard file format of Hall *et al.* (1991), and for the crystallographic information framework, a broader system of exchange protocols based on data dictionaries and relational rules expressible in different machine-readable manifestations, including, but not restricted to, crystallographic information file and XML (Bray *et al.*, 2004). Fig. 1 is a short snippet of `1zrt.cif`.

The very large sizes and short data collection times of raw synchrotron data images make pure ASCII text formats less desirable than binary formats. Since CIFs are pure ASCII text files, a separate binary format had to be defined to allow the combination of pseudo-ASCII sections and binary data sections to handle raw synchrotron data images within the context of CIF. CBF and imgCIF (Bernstein & Hammersley, 2005) are two aspects of the same format. The binary file format is the crystallographic binary file (CBF). The ASCII sections are very close to the CIF standard but must use operating-system-independent 'line separators'. imgCIF is also the name of the CIF dictionary (Hammersley *et al.*, 2005) that contains the terms specific to describing the binary data. The imgCIF dictionary is layered on the macromolecular CIF (mmCIF) dictionary (Fitzgerald *et al.*, 2005).

3. *VCIF2* overview

The default input for *VCIF2* is `stdin` but files can be specified with `-i` filename. Both CIF and CBF formats are supported. The program

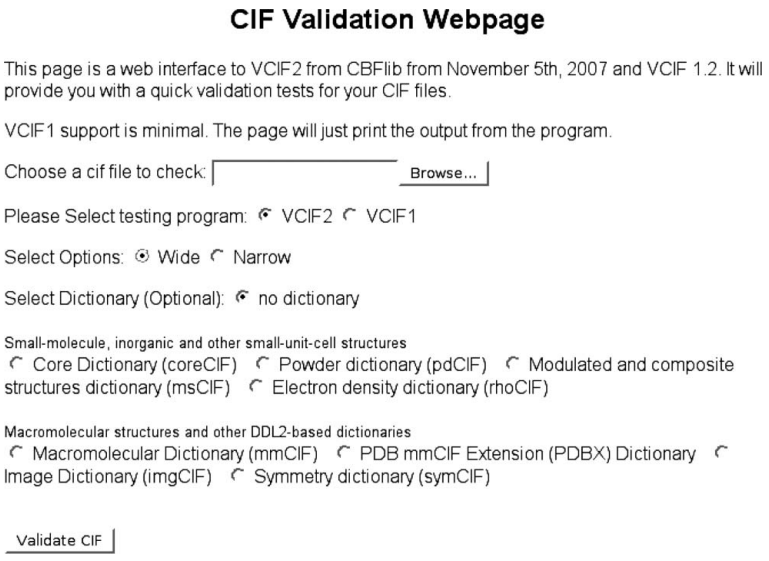

Figure 2
VCIF2 web interface at http://www.vcif.org.

prints the output file to `stdout` or to a file specified with `-o` filename. If base64 or quoted-printable encoding is used, the output file will be in CIF format, otherwise CBF. All errors and warnings are sent to `stderr`. If long-line CIFs are being processed the `-w` option is required in order to avoid 'over line size limit' warnings and to output wide lines instead of folding them.

VCIF2 also supports dictionary validation. A dictionary is specified with the `-v` option.

On read, the parser checks every token (*e.g.* word, punctuation *etc.*). First it does a syntax check and then it performs more in-depth validation, such as dictionary and parent–child relationships.

An example for validating a wide-line CIF against the PDBx dictionary would be `vcif2 -w -v mmcif_pdbx.dic -i 1zrt.cif /dev/null`, where `mmcif_pdbx.dic` is the dictionary, `1zrt.cif` is the CIF file to be validated and `/dev/null` means the output will be discarded (for Unix machines).

4. *VCIF2* web interface

Because of the popularity of the World Wide Web we created a web interface to *VCIF2*. This simplifies the process of using the program. The user is required to have a web browser and a CIF file for validation. This web interface can be accessed *via* the *CIF Validation Webpage* (Todorov, 2006). Currently supported dictionaries are coreCIF, pdCIF, msCIF, rhoCIF, mmCIF, PDBx, imgCIF and symCIF.

The input file is specified *via* an open file dialog and the rest of the options are radio buttons on the web page (Fig. 2). The output is generated after the Validate button is pressed and will contain the original output of *VCIF2*. The line numbers where errors were detected will be hyperlinks to sections after *VCIF2*'s output that correspond to the detected error. Each corresponding section provides five context lines from the validated CIF file with the problem line in bold in the middle. Additionally, if common mistakes are detected, suggestions are provided in the same section. See Figs. 3 and 4 for examples of a string error and two quote errors, respectively, detected by *VCIF2*.

5. Implementation

VCIF2 has been embedded in an existing utility called *cif2cbf* which is part of the CBF library (CBFlib; Ellis & Bernstein, 2001, 2005). The program name *VCIF2* is simply an alias for *cif2cbf* with appropriate command line options. The majority of it is written in standard C, with small parts making use of Fortran and the yacc parser. The web interface uses standard HTML forms with a php script in the back end for parsing and executing the *VCIF2* binary on our server. When *VCIF2* validates against a dictionary, the dictionary populates a database-like table, represented as a CIF file in memory. After the lexical and parser validations are performed, the input CIF is checked against the dictionary for validity.

For binary synchrotron imgCIF data, the program checks the validity of the header tags and the ranges of their values, and the validity and checksum of the MIME header of the actual binary image, but the image itself is not validated, other than for the checksum and size.

6. Distribution

VCIF2 is called *cif2cbf* in CBFlib and is located in its examples folder. Current development of CBFlib is being carried out on our GForge server (Arcib Laboratory, 2006*a*). Complete developer or binary kits can be found *via* the file release system on the project's website (Arcib Laboratory, 2006*b*) or *via* CVS (Arcib Laboratory, 2006*c*).

```
VCIF2 Output:

CBFlib: error input line 301 (6) -- value without tag
...
The error message was: error input line 301 (6) -- value without tag

    299 _refine.pdbx_ls_cross_valid_method                THROUGHOUT
    300 _refine.details

    301 ;IN CHAINS D AND Q THE LOOP 161 TO 179 IS POORLY ORDERED AND
SEQUENCE

######################################################################
# Could line 301 be a quoted string with unintended leading blank?   #
######################################################################

    302 COULD NOT BE ASSIGNED. FURTHERMORE IT WAS TRACED WITH FEWER RESIDUES
    303 THAN ARE ACTUALLY PRESENT. RESIDUE 167 COULD BE LOCATED BECAUSE IT
```

Figure 3
`1zrt.cif` string error.

```
VCIF2 Output:

CBFlib: warning input line 5 (1) -- ended before end of single-quoted string
CBFlib: warning input line 10 (1) -- ended before end of single-quoted string
 Time to read input_cif: 0.420s
The error message was: warning input line 5 (1) -- ended before end of single-quoted string

      3 loop_
      4 _audit_author.name
      5 'Berry, E.A."
      6 'Huang, L.S.'
      7 'Saechao, L.K.'
The error message was: warning input line 10 (1) -- ended before end of single-quoted string

      8 'Pon, N.G.'
      9 'Valkova-Valchanov, M.'
     10 'Daldal, F
     11 #
     12 _pdbx_dat abase_st at us. st at us_code                REL
```

Figure 4
`1zrt.cif` single and double quote errors.

The latest testing version of CBFlib is in the CVS repository under module name `CBFlib_bleeding_edge`. The latest stable version can be found in the CVS with module name `CBFlib_latest_stable`. The web interface code, along with dictionaries and *VCIF2* binaries for Linux, can be found in the CVS under module name `CBFlibHTML` or at the download page (Arcib Laboratory, 2006*d*).

For more information, readers are invited to send e-mail to yaya@bernstein-plus-sons.com or terahz@geodar.com

APPENDIX *A*
Glossary

The following is a glossary of terms used in this paper.

ASCII	American standard code for information interchange
base64	A method of encoding binary data sent as an attachment through e-mail
CBF	Crystallographic binary file
CIF	Crystallographic information file
CVS	Concurrent versioning system
imgCIF	Image-supporting crystallographic information file
IUCr	International Union of Crystallography
mmCIF	Macromolecular crystallographic information file
msCIF	Modulated and composite structures dictionary
PDBx	Protein Data Bank exchange directory
pdCIF	Powder dictionary
php	PHP hypertext preprocessor
`stderr`	The standard error stream in Unix
`stdin`	The standard input stream in Unix
`stdout`	The standard output stream in Unix
rhoCIF	Electron density dictionary
symCIF	Symmetry dictionary
yacc	Yet another compiler, the standard parser generator on Unix systems

This work was supported in part by the International Union of Crystallography, the US National Science Foundation and the US Department of Energy.

References

Arcib Laboratory (2006*a*). *GForge CBFlib Project*, http://blondie.dowling.edu/projects/cbflib/.
Arcib Laboratory (2006*b*). *CBFlib Releases Webpage*, http://blondie.dowling.edu/frs/?group_id=10.
Arcib Laboratory (2006*c*). *CBFlib CVS Repository*, http://blondie.dowling.edu/scm/?group_id=10.
Arcib Laboratory (2006*d*). *CIF Validation Webpage Download*, http://www.vcif.org/get.html.
Bernstein, H. J. & Hammersley, A. P. (2005). *International Tables for Crystallography*, Vol. G, *Definition and Exchange of Crystallographic Data*, pp. 37–43. Heidelberg: Springer.
Bray, T., Paoli, J. & Sperberg-McQueen, C. M. (2004). *World Wide Web Consortium*, February issue, http://www.w3.org/TR/2004/REC-xml-20040204.
Ellis, P. J. & Bernstein, H. J. (2001). *CBFlib: An API for CBF/imgCIF Crystallographic Binary Files with ASCII Support*, http://www.bernstein-plus-sons.com/software/CBF.
Ellis, P. J. & Bernstein, H. J. (2005). *International Tables for Crystallography*, Vol. G, *Definition and Exchange of Crystallographic Data*, pp. 544–556. Heidelberg: Springer.
Fitzgerald, P. M. D., Westbrook, J. D., Bourne, P. E., McMahon, B., Watenpaugh, K. D. & Berman, H. M. (2005). *International Tables for Crystallography*, Vol. G, *Definition and Exchange of Crystallographic Data*, pp. 295–443. Heidelberg: Springer.
Hall, S. R., Allen, F. H. & Brown, I. D. (1991). *Acta Cryst.* A**47**, 655–685.
Hall, S. R., Westbrook, J. D., Spadaccini, N., Brown, I. D., Bernstein, H. J. & McMahon, B. (2005). *International Tables for Crystallography*, Vol. G, *Definition and Exchange of Crystallographic Data*, pp. 20–36. Heidelberg: Springer.
Hammersley, A. P., Bernstein, H. J. & Westbrook, J. D. (2005). *International Tables for Crystallography*, Vol. G, *Definition and Exchange of Crystallographic Data*, pp. 444–458. Heidelberg: Springer.
McMahon, B. (1998). *VCIF: a utility to validate the syntax of a crystallographic information file*, http://www.iucr.org/iucr-top/cif/software/vcif/index.html.
McMahon, B. (2005). *International Tables for Crystallography*, Vol. G, *Definition and Exchange of Crystallographic Data*, pp. 499–525. Heidelberg: Springer.
Todorov, G. (2006). *CIF Validation Webpage*, http://www.vcif.org.