

Writing CIF Software

Some points to consider



Available tools and libraries

<https://www.iucr.org/resources/cif/software>

The screenshot shows the IUCr website page titled "SOFTWARE FOR CIF AND STAR". The page features a navigation menu at the top with links for "THE IUCR", "NEWS", "PUBLICATIONS", "PEOPLE", "RESOURCES", "EDUCATION", and "OUTREACH". Below the navigation, there are sub-links for "specification", "documentation", "software", "concepts", "diff", "dictionaries", "checkCIF", and "cif2".

The main content area is titled "SOFTWARE FOR CIF AND STAR" and includes a brief description: "The software listed on this page includes copies of, or links to, many of the programs and libraries available for use with CIF and STAR files. They include archival copies of software that is no longer under active development, and a subset of the many projects that are currently under active maintenance."

There are two key sections: "Browse by category:" and "Keys:".

Browse by category:

1. General STAR File utilities
2. CIF validation
3. Software libraries
4. File manipulation programs
5. coreCIF applications
6. pCIF and mcCIF applications
7. mcCIF applications
8. mcCIF/CIF applications
9. Editors with graphical user interfaces
10. Visualization tools
11. Rapid development tools
12. Obsolete or early development software
13. Other

Keys:

- Archival or reference copy of software available from this server
- Link to external site supplying software
- Link to web service

On the right side of the page, there is a sidebar with a search bar and a list of categories: "Specifications", "Documentation", "Software", "CONCEPTS", "Dictionary definition languages", "CIF dictionaries", "checkCIF", and "CIF version 2".

At the bottom right, there is a red banner for the "INTERNATIONAL TABLES FOR CRYSTALLOGRAPHY" (Volume C: Dictionary and Tables of Crystallographic Data), edited by S. R. Hall and B. Muirhead.



##

Writing CIF Software



Available tools and libraries

<https://mmcif.wwpdb.org/docs/software-resources.html>

PDB/mmcif Software Resources

Software Libraries and Tools

- mmCIF-RCSB Python based mmCIF Core Access Library replacing CCP4/RSB OBI wrapper
- CCP4 MMIO - MMIO is a macromolecular coordinate library, supporting CCP4 applications such as REFMAC and COOT
- REFMAC C++11 macromolecular library with Python and Fortran bridge
- BioCif C++11 general purpose mmCIF and PDB parser with strict implementations of the mmCIF, DDL and PDB/mmcif dictionary
- cif tools a series of tools for the manipulation, analysis and conversion of mmCIF files (depends on libcif++)
- libcif++ implements reading and writing of CIF files (specification) as well as their efficiently executed counterpart, called BinaryCIF
- libmmCIF Python package to parse PDB, MMIO and mmCIF formats
- libmmCIF C++ library providing C/C++ support and dictionary validation
- libcif and libcif++ C++ Python library used by Phenix for reading and writing CIF files, including dictionary validation
- CIFWRITE C/C++ RCSR PDB C++ library for accessing and checking PDB/mmcif data files and dictionaries with supporting tutorial and examples
- CIFWRITE C/C++ Python wrapper: BoostPython wrapper for RCSR PDB C++ CIFWRITE C/C++ library
- Python PDB - RCSR PDB Python light weight readwrite tools for PDB/mmcif files and dictionaries with examples and tutorials
- PDBtoCIF - PDB Python package to work with mmCIF formatted files including tutorial and examples
- mmCIF - A fast C++ CIF and mmCIF parser
- CIFLib - Library of ANSI-C functions providing a simple mechanism for accessing Crystallographic Binary Files (CIF files) and Image-supporting CIF (imgCIF) files
- mmCIF - Python macromolecular toolkit (including support for mmCIF)
- BioJava mmCIF - java input and output methods for mmCIF
- and a tutorial for the protein structure modules. Both support obsolescence, fully supporting BinaryCIF as well
- BioPython - Python toolkit for computational biology including support for mmCIF
- PyCifRW Python CIF read and write methods
- Star Tools STARcif - Local analyzer for STARCIF mmCIF data (including a JAR implementation)
- Fast STAR (mmCIF) Parser Several object-oriented Perl modules to parse mmCIF data files
- Jmol2CIF Open source C++ program for opening Jmol to PDB format

Visualization Tools Supporting PDB/mmcif

- Jmol a modern web-based open source toolkit for visualization and analysis of large scale molecular data
- PDBtoCIF - wrapper around Jmol
- RCSR PDB to CIF - wrapper around Jmol
- PyMol a user-sponsored molecular visualization system on an open source foundation
- Jmol Java macromolecular visualization tool supporting mmCIF
- CIF2D Chemical grid-generation molecular visualization system, Jmol and within mmCIF
- CIF2D Chimera molecular visualization system, mmCIF
- ChimeraX Open source version of Chimera with mmCIF support
- Chimera Crystallographic Object Oriented Toolkit
- CCP4/RSB CCP4 Molecular Graphics Package

Software Tools Developed at wwPDB

- PDB_EXTRACT Tools and examples for extracting mmCIF data from structure determination applications
- MMIO An application for processing and analyzing of macromolecular structure data including many format conversion features
- Jmol An Applet A format conversion tool for structure factor files that supports mmCIF data storage file formats



#/#

Writing CIF Software



Available tools and libraries

- **CIF API** A C API and reference implementation for CIF 2.0 (and earlier) https://github.com/COMCIFS/cif_api
- **PyCifRW** CIF/STAR file support for Python <https://pypi.org/project/PyCifRW/>
- **CrystalInfoFramework.jl** Julia tools for working with CIF, including reading CIF1 and CIF2 data files (includes mmCIF files from PDB) <https://github.com/jamesrhester/CrystalInfoFramework.jl>
- **JsCifBrowser** A browser-based interface to DDLm dictionaries with dREL executable algorithms <https://github.com/COMCIFS/JsCifBrowser>
- **COD::CIF::Parser** An error-correcting CIF parser for the Perl language (part of cod-tools) <http://wiki.crystallography.net/cod-tools>
- **pycodcif** Python interface to COD CIF parser language (part of cod-tools) <http://wiki.crystallography.net/cod-tools>

In development

- IUCr CIF dictionary browser (<https://www.iucr.org/resources/cif/dictionaries/browse>) and **publicIF** (CIF1.1 version currently at <https://journals.iucr.org/services/cif/publicif/>)



#/#

Writing CIF Software



Advantages of dictionary-based software

1. Data name recognition
2. Alternative data name recognition (aliases)
3. Type and range validation
4. Referential integrity
5. Application of dREL methods



#/#

Writing CIF Software



Best practice

- Validate against appropriate dictionary version
(`_audit_conform.dict_name`, `_audit_conform.dict_version`,
`_audit_conform.dict_location`)
- Consult list of registered prefixes if unexpected data names
found (<https://www.iucr.org/cgi-bin/cifreserve.pl>)
- Check `_audit.schema`



#/#

Writing CIF Software



Advanced best practice

- Error trapping and handling
 - Notify user (notice, warning, error)
 - Auto-fix (how?)
- Handling of duplicate data names in same data block
 - Exact duplicates are clear error; die or accept if values identical?
 - Aliases: more lenient? (Could be strategy to help legacy software)



#/#

Writing CIF Software



Things to take particular care about

- Ambiguity of data values that look like numbers.
- How to handle quoted numeric strings (*e.g.* ' 12 ')?
- Quoting styles (single vs double quotes, triple quotes, semicolons).
- Handling of 'scalar' data [*e.g.* adopt the CIF API approach of storing these as single-packet loops
 - But how to present them as unlooped?
 - What to do if a (normally) multi-packet loop only has one packet?].
- Long line folding.



#/#

Writing CIF Software



Handling CIF2 and CIF1 syntax

- NB The presence of ‘dotted’ data names (`_cell.volume`) is not a robust indicator that a file is CIF2 (likewise all-underscore data names do not guarantee that it is CIF1).
- Initial ‘magic’ string `##CIF_2.0` is mandatory for CIF2.
- Handling different character encodings (ASCII vs Unicode).
- Conversion between different quoting mechanisms.
- Semantic nuances considered in detail in Vaitkus, A., Merkys, A. & Gražulis, S. (2021). *Validation of the Crystallography Open Database using the Crystallographic Information Framework*. *J. Appl. Cryst.* **54**, 661–672.



#/#

Writing CIF Software

